

2011.04

PROGRAMMER

程序员

HTML



来临!

26 2011 程序员薪资 调查报告

剖析中国软件开发者的普遍生存状态，
定位您在软件产业内的坐标。

68

facebook

工程管理揭秘

让亲身实践者执行工作流程

70 模型驱动开发

在RUP与Agile之间找到平衡点

78 架构师接龙

新浪 杨海朝 vs 去哪儿网 孙立

81 前 eBay 高级副总裁

Marty Cagan

谈产品管理与项目管理、产品
设计的关系

84 创业团队文化谈

88 硅谷创业之父

Paul Graham

梦寐以求的编程语言

115 volatile

与多线程的那些事儿

120 maven 十年

ISSN 1672-3252



邮发代号：2-665 定价：15元

www.programmer.com.cn www.csdn.net

特别感谢

(排名不分先后, 以姓氏首字母为序)

Contributors



崔康

感谢 IBM 研发中心工程师崔康, 介绍 HTML5 在浏览器兼容方面的表现, 详见“封面报道”栏目。



冯勇

感谢 前天涯高级架构师冯勇, 介绍 NoSQL 数据库 memlink 概况, 详见“产品报道”栏目。



高寒蕊

感谢 Google 开发技术推广工程师高寒蕊, 分享 HTML5 相关背景知识, 详见“封面报道”栏目。



黄后锦

感谢 酷 6 开发部项目经理黄后锦, 分享 HTML5 在视频网站的应用, 详见“封面报道”。



蒋宇捷

感谢 腾讯研究院前端负责人蒋宇捷, 介绍 HTML5 相关应用介绍, 详见“封面报道”栏目。



Adrian Johnston

感谢 微软商务解决方案事业部亚太区总经理 Adrian Johnston 为本刊提供重要观点支持, 详见本期“高端视点”栏目。



李冬梅

感谢 北大附中信息技术特级教师李冬梅接受本刊记者采访, 详见本期报道栏目。



刘龙龙

感谢 台湾铭传大学国际学院教授刘龙龙, 阐述软件复用对程序员技能的影响, 详见“评论”栏目。



楼天城

感谢 中国大学生计算机编程第一人楼天城接受本刊记者专访, 详见“程序人生”栏目。



马如悦

感谢 百度基础架构部高级工程师马如悦对“论文研读”栏目的支持。



潘加宇

感谢 UMLChina 首席专家潘加宇, 分享使用工具的心得体会, 详见“我的工具箱”栏目。



孙立

感谢 去哪儿网高级系统架构师孙立为本刊撰文, 分享架构方面的经验, 详见“架构师接龙”栏目。



王皓

感谢 虾米网总经理王皓为本刊撰文, 详见“一分钟先生”栏目。



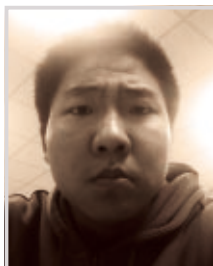
王建硕

感谢 百姓网 CEO 王建硕为本刊撰文, 详见“一分钟先生”栏目。



王子龙

感谢 百度电子商务事业部前端工程师王子龙, 介绍 HTML5 激动人心的未来发展, 详见“封面报道”栏目。



魏子钧

感谢 惠普 Web 开发工程师魏子钧, 介绍 HTML5 在 2D 动画方面的应用, 详见“封面报道”栏目。



萧涵忆

感谢 微软 MVP 萧涵忆接受本刊记者采访, 详见本期报道栏目。



许晓斌

感谢 Sonatype 公司 Maven 技术专家许晓斌, 介绍 Maven 的相关知识, 详见“产品报道”栏目。



杨栋

感谢 百度分布式研发工程师杨栋为本刊撰文, 分享 Facebook 数据库技术, 详见“技术”栏目。



杨丰盛

感谢 佳玩无线联合创始人、《Android 应用开发揭秘》一书作者杨丰盛为本刊撰文, 详见“移动专栏”。



杨海朝

感谢 新浪首席 DBA 杨海朝为本刊的支持, 详见“架构师接龙”栏目。



于敦德

感谢 途牛旅游网 CEO 于敦德为本刊撰文, 详见“一分钟先生”栏目。



张勇

感谢 驴妈妈 CTO 张勇接受本刊记者采访, 详见本期“对话 CTO”栏目。



赵威

感谢 天涯社区系统平台部数据系统架构师赵威, 介绍 NoSQL 数据库 memlink 概况, 详见“产品报道”栏目。

开放与封闭

近来国内许多公司都开始意识到，需要效仿世界领导性的公司，建立开放的生态系统。可是，为什么要开放？又应该怎么开放？这些都是大话题，值得深入探讨。

我们必须认识到，开放并非无条件的褒义词。在经典著作《反模式》中，有一个著名的反模式叫做“委员会设计”：由于有太多的人参与设计，意见纷繁，利益纠缠，最终设计变得过度复杂、含糊不清，根本无法有效地真正实现。也就是说，过度开放无法产生好的设计。历史上，有太多技术和标准都因为“委员会设计”，最后逐渐演化为垃圾场：ATM、ISO OSI网络模型、CORBA、UML、JavaEE、SQL……

开放也不是成功的代名词。IBM PC通过开放体系架构虽然很轻松地打败了Apple等早期风光的厂商，但最终胜利的果实却属于Windows+Intel联盟，以及HP/Compaq和Dell这些后来者。而一直以大幅度开放享誉世界的Twitter始终没有找到太好的赢利方式，最近不得不一改语调，明确宣布不再欢迎第三方开发者开发新的客户端，引起众多争议。

与此相反的是，Apple自己做CPU、自己做操作系统、自己做各种软硬件、自己全面管理和运营应用商店，甚至自己负责产品销售，并用近乎铁腕的风格管控着整个生态链，对任何潜在的竞争者或者山头都毫不客气地痛下杀手……通过这种独特的封闭体系，创造了本世纪以来最绚丽的科技奇迹。

当然，如果用放大镜观察细节，我们可以发现，Apple其实是很善于利用开放技术的公司。操作系统的内核是BSD和Mach，CPU是ARM架构，iPod的软硬件和设计最初都是由外部公司完成的……而最重要的，则是Apple使应用商店成为新的开放模式，为广大开发人员和内容创作者创造了广阔的舞台，也为自己带来巨大的竞争优势。同时，Twitter的开放性仍然是毋庸置疑的，这个微博平台上的75万个应用，不仅远远超过Apple App Store，也构筑了后来竞争者很难超越的门槛。IBM的开放对整个社区的意义非常重大，而它自己的失败肇因恰恰是在总线之战中采取了封闭专有的路线。

本期我们关注的是HTML5。有意思的是，HTML5的历史恰恰提供了研究开放与创新的一个绝佳案例。1997年W3C决定将HTML变身为一组XML标签集，但XHTML的发展非常缓慢，远远不能满足日益兴起的Web应用的需要，而向后兼容性问题使一些实战派的浏览器开发者们开始意识到沿着这个方向将走向死胡同。2004年6月，Opera和Mozilla首先发难，建议W3C改弦易辙，回到HTML的基础上演进。但是提案未获通过。这次圣荷塞会议促成了HTML5的摇篮WHATWG诞生。3年后，W3C主导的XHTML 2遥遥无期，而WHATWG却进展顺利，并被W3C再次接纳，成为下一代核心Web标准的基础。

值得思考的是WHATWG这个机构的组织方式。它非常开放：任何人都可以通过加入邮件列表成为贡献者；同时也很封闭：能够决定标准编辑人选的是一个邀请制的小组。同时，它也为“委员会设计”反模式提供了一个很好的解决范例。



邮件：liujiang@csdn.net

新浪微博：@刘江CE

欢迎大家交流反馈，欢迎投稿、批评、建议和挑错

HTML



来临!

40 封面报道 Cover Story

作为下一代互联网标准，HTML5的地位举足轻重。无论你是重视互联网产品，还是关心用户体验，或者专注于前端开发，HTML5都是不可错过的技术热点。本期封面报道，聚焦HTML5技术，邀请从事一线开发的工程师，揭秘HTML5，展现HTML5不可限量的应用前景。

- 42 HTML5技术应用分享
——第八期TUP系列活动报道
- 44 浏览器对HTML5的兼容性分析

- 49 激动人心的HTML5之美
- 53 清音油言
——献给HTML5

- 57 HTML5 2D游戏开发实例剖析
- 61 HTML5未来展望

固定专栏 Columns

- 7 名人堂 Hall of Fame
- 8 外刊速递 Abroad Media
- 12 微博 Micro-Blogging
- 16 程序天下事 Technical News
- 134 十年前 Review
- 136 漫画与幽默 Humor

高端视点 Leaders' Viewpoint

- 14 成功实施CRM的关键因素

报道 Report

- 26 2011程序员薪资调查报告
- 30 2011推荐系统论坛随记
- 32 NoSQL利器MongoDB驾临中国
——MongoDB开发者会议引爆京城
- 34 微软MVP萧涵忆：WP7将和Android、iPhone三足鼎立
- 35 学生兴趣是信息技术教育的切入点
——北大附中信息技术特级教师李冬梅专访

对话CTO CTO Columns

- 36 互联网的将来：现实与虚拟结合
——记CSDN CTO俱乐部3D互联网专业委员会首次线下活动
- 38 分兵策略应对高速发展

程序人生 Coding Life

- 66 我的编程竞赛之路
——中国大学生计算机编程第一人楼天城访谈编程奇才楼天城讲述他的精彩程序人生。

工程管理 Engineering Management

- 68 让亲身实践者执行工作流程
Facebook前工程总监黄易山 (Yishan Wong) 撰写了一系列文章，很好地总结了Facebook卓越研发文化中的宝贵经验。本刊将陆续连载这一系列，本文是第二篇。

软件工程 Software Engineering

- 70 模型驱动开发
——在RUP与Agile之间找到平衡点
本文介绍了集统一软件开发和敏捷开发方法优点于一体的新型软件开发方法——模型驱动开发。

74 ThoughtWorks技术雷达

1月份最新发布的《技术雷达》认为以下几个技术趋势越来越明显：对软件持续交付及其方法的关注；云端产品的多样性和深度；更高效地运用基本的互联网技术。

架构 Architecture

- 78 新浪杨海朝 VS. 去哪儿网孙立
在架构设计中如何应对不断出现的超过预估的数据量和访问量？分布式的数据存储层如何结合应用特点选择合适的存储方式？分布式数据库中如何考虑一致性、分区策略和延时问题？两大高手，隔空过招，精彩内容，尽在本期架构师接龙。

TUP专栏

- 81 产品管理与项目管理、产品设计的关系
Marty Cagan是享有世界声誉的产品管理专家，曾经担任网景副总裁、eBay产品管理及设计高级副总裁。本文是他回顾自己二十多年来从事软件产品管理工作的总结和经验分享，谈到了产品管理、项目管理、产品设计的关系。

一分钟先生 Mr. One Minute

84 互联网创业团队环境文化建设谈

技术 Technology

88 梦寐以求的编程语言（上）

Paul Graham是公认的“硅谷创业之父”，对于许多技术问题，他有自己独到的见解。在本文中，他分享了何谓理想的编程语言。文章分两期刊登，第二部分将刊登于5月刊。

92 RCFile——用于Facebook数据仓库的高效存储结构
本文介绍了Facebook公司数据分析系统中的RCFile存储结构，该结构集行存储和列存储的优点于一身，在MapReduce环境下的大规模数据分析中扮演重要角色。

96 浅析.NET应用程序的内存问题

曾经困扰很多开发人员的“内存泄漏”问题目前仍值得我们关注，本文介绍了一些在.NET世界里的典型内存问题及其解决方案。

100 调整Netfilter内核模块以限制P2P连接

104 Azure和Bing Maps API示例经验分享

115 volatile与多线程的那些事儿

在进行多线程编程时，我们需要谨慎使用volatile关键字，本文分析了C/C++和Java多线程程序中使用volatile关键字的一些例子，帮助大家更好地理解和使用volatile。

调试之剑 Debugging Sword

108 在调试器中看Win7打电话回家（下）

移动专栏 Mobile

112 Android自动化测试之道

随着越来越多的手机厂商、运营商、开发商加入Android阵营，带来了各种美化和定制的Android系统，同时Android Market的应用数量增长迅速。这都离不开测试，Android自动化测试的需求愈加迫切。

工具点评 Tools Reviews

118 我的工具箱

产品报道 Products Report

120 Maven十年，来认识下

123 非关系型数据库memlink浅析

产品推荐 Products Recommendation

126 Geek产品

128 新产品新工具

图书 Books

130 新书上架

评论 Comments

132 从软件复用看现代程序员的竞争力

133 长翅膀的不一定是天使

主管：中国社会科学院

主办：中国社会科学院文献信息中心

出版：《程序员》杂志社

网址：http://www.programmer.com.cn

国际刊号：ISSN 1672-3252

国内刊号：CN11-5038/G2

邮发代号：2-665

广告经营许可证号：京东工商广字0188号

总编：黄长著 Editor-in-chief: Huang Changzhu

社长/常务副总编：张悦校 President: Zhang Yuexiao

副社长：蒋涛 Vice President: Jiang Tao

编委会：黄长著 张悦校 陈洋彬 蒋涛 曾登高 刘江

Editorial Member: Huang Changzhu Zhang Yuexiao Chen Yangbin Jiang Tao
Zeng Denggao Liu Jiang

执行主编：孟迎霞 Executive Editor-in-chief: Meng Yingxia

编辑部主任：常政 Director: Chang Zheng

责任编辑：董世晓 高松 郑柯

Editors: Dong Shixiao Gao Song Zheng Ke

特邀编辑：方梁 高昂 赵健平 吕娜 卢鹤翔

Contributing Editors: Fang Liang Gao Aang Zhao Jianping

Lv Na Lu Dongxiang

美术设计：纪明超 Art Designer: Ji Mingchao

美术编辑：林象海 Art Editor: Lin Xianghai

流程编辑：陈秋歌 Coordinator: Chen Qiuge

Tel: 010-64351458

Email: editor@csdn.net

发行部 Distribution Dept.010-64351431

Email: sales@csdn.net

广告总代理：北京创新乐知广告有限公司

Sole Advertising Agency: Beijing CSDN Co.,Ltd

Tel: 010-64376055

Email: ad@csdn.net

Marketing Dept: 010-51661202 (ext 149)

Email: market@csdn.net

读者服务部

Readers service Dept.

网上订购：http://dingyue.programmer.com.cn/

读者信箱：reader@csdn.net

地址：北京市朝阳区广顺北大街33号院1号楼福码大厦B座12层

Address: B-12th Floor Fairmont Tower NO.33 Guangshun North street,
Chaoyang District, Beijing

邮政编码：100102

电话：010-64351436

传真：010-64348545

法律顾问：北京中润律师事务所 王杰

Law Consultant: Beijing Hengsheng Lawyer Firm

印刷：北京盛通印刷股份有限公司

Print: Beijing Shengtong Printing Co., Ltd.

出版日期：每月1日

Publication Date: the first day per month

零售价：RMB 15.00元 新台币 390元 HK \$ 35.00（港、澳）

US \$ 9.00（海外）

Retail Price: RMB 15, NT\$390, HK \$ 35.00, US \$ 9.00

本刊文章版权所有 未经许可不得转载

发现装订错误或缺页，请将杂志寄回本刊读者服务部，即可得到调换。

数值计算先驱

——James Hardy Wilkinson

文 / 苏柳

上一期名人堂中，介绍了Wilkes和他的EDSAC计算机。其中说到，EDSAC项目是英国第三个电子计算机项目，与此同时还有图灵领导的ACE项目。

1945年，图灵刚刚放下“二战”的情报破译工作，就立即设计了这种全新的电子计算机，并向政府申请资助。图灵提交了一份非常详细的设计报告，然后一直在国家物理实验室埋头编写指令表（相当于我们现在编写函数库），并改进机器的设计。

经过很长时间的讨论，政府批准了这个项目，由达尔文爵士领导，图灵担任总设计师。达尔文为这个项目另外调派了两位科学家，其中之一，就是本文的主角，James Hardy Wilkinson。

1919年9月27日，Wilkinson在英国肯特郡的斯特洛特出生，16岁时被剑桥大学破格录取，19岁毕业获得最高荣誉奖章。刚毕业，二战就爆发了，于是Wilkinson进入军备研究所，研究与弹道有关的问题，在这里，他对数值计算产生了浓厚的兴趣。

战后，Wilkinson来到国家物理实验室协助进行一项数值计算工作，他看到图灵的ACE项目，立刻被吸引住了，并且留在这里工作。后来这个项目发生了一些人事上的矛盾，把图灵给气跑了（这事儿我们上期也提过），于是Wilkinson就接管了整个ACE原型机的建造工作。

Wilkinson接手之后，总结了前一阶段的教训，果断采取了两项措施，

一是与工程实践团队加强联系，改变过去的僵局；二是把ACE的设计方案回档到第五版（图灵走的时候，已经做完了第八版设计），把延迟线的数量减少到32根。

1950年5月10日，ACE原型机第一次正式试运行成功。1950年11月，NPL举行了隆重的发布仪式，随后由EEC公司量产了大约30台，其商业名称为DEUCE。DEUCE和Wilkes的EDSAC计算机一起，使英国的计算机技术跃居世界领先水平，与美国不相上下。第一台ACE原型机至今仍在伦敦的肯辛顿科学博物馆里展出，有机会的读者可以前去参观。

除了ACE项目之外，Wilkinson在数值计算方面也做出了先驱性的贡献。1960年，Wilkinson提出了“向后误差分析法”，这是误差理论中最基本的方法之一。向后误差分析是一种先验估计，假设一个计算结果 x ，是由已知量 $a_1, a_2, a_3, \dots, a_n$ 通过基本运算确定的，也就是：

$$x = f(a_1, a_2, a_3, \dots, a_n)。$$

由于数字计算机会产生舍入误差，实际表示出的值 x' 与准确值 x 会有不同。向后误差分析法把舍入误差与已知量 $a_1, a_2, a_3, \dots, a_n$ 的某种扰动联系起来，引入扰动量 ε_i ，得到 x' 的表达式：

$$x' = f(a_1 + \varepsilon_1, a_2 + \varepsilon_2, a_3 + \varepsilon_3, \dots, a_n + \varepsilon_n)$$

然后再推出这些 ε_i 的界，并以此推算最后的舍入误差 $x' - x$ 的界。最常见的例子，就是浮点数的计算。我们大家都知道，浮点数的机器表示为 $f = p * m^e$ （其中 p 是尾数， m 是基值， e 是阶码）。

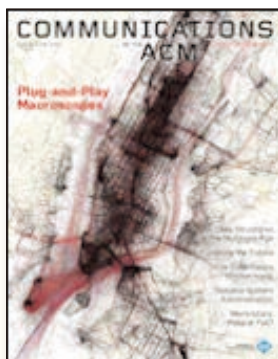


这就是典型的由多个已知量共同确定的数值，而且它是有误差的（在固定的位数和基底下，总是有一部分浮点数无法表示）。在这种情况下，如果需要分析其误差，那么向后误差分析法就是最常用的方法。

在那个年代，数值计算是计算机的主要用途，所以这一类算法具有非常重要的意义。

1970年，ACM将图灵奖授予Wilkinson，以表彰其在ACE计算机项目、数值计算，尤其是向后误差分析法方面的贡献（笔者个人认为，如果图灵此时还活着，考虑到他当初的ACE设计方案，他绝不会允许Wilkinson获得图灵奖）。同年，Wilkinson还获得了冯·诺伊曼奖。

Wilkinson一直为国家物理实验室工作了34年，并在1980年退休后，担任斯坦福大学客座教授。1986年10月5日，Wilkinson在英国特丁顿的家中因病去世，享年67岁。P



Communications of ACM

2011.03

Twitter上发布的信息包罗万象。其中有很多信息噪音，但总的来说，Twitter为人类情绪、思维和社会行为的研究提供了海量的信息库。

《Twitter：媒体和信息源》一文介绍了研究人员如何设计算法在这些喋喋不休中寻找特定的关键词和精确的消息来源，来分析大众情绪，跟踪政治动态，还有流感爆发的趋势。从Twitter分析中可以获得出人意料精确的结果。

研究人员使用关键词，在Twitter中寻找流感爆发的趋势。虽然专业医疗机构的统计更精确，但也更耗时耗力。Twitter分析的实时性好、信息收集成本低，可以提供早期的流感预警，辅助医疗机构做出决策。

Twitter流感分析和Google的“流感趋势”（flu trends）工作原理类似。但Twitter信息中包含的字数更多，信息更丰富，比起Google中只有两三个词的搜索引擎关键字，更有优势。而且在对算法进行训练优化后，可以获得更好的信息分类效果。

另外，Twitter分析也可以作为自然现象监控的有益补充，比如空气质量检测、地震预报，等等。甚至还可以帮助研究人员排除地震误报等情况。

Twitter分析在社会学领域也有很多用武之地。研究人员使用心理学工具对Twitter消息进行大众情感分析。这种做法有一定局限性，比如说Twitter信息中包含很多噪音，而且有些词汇的含义可能被误解，但是因为消息基数很大，误差会相应减小；另外情绪的互相感染也会影响统计结果；还有采样偏差的问题。

Twitter分析也可以用于进行社会调

查，弥补传统调查方法的缺点。人们对人们自愿发布的Twitter信息进行分析，可以避免调查问卷问题中暗含倾向性的误导。也可以代替费时费力的传统方法对某些以前难以实施的社会学问题进行调查。

另外，Twitter也是研究信息传播方式的好场所。

Twitter信息快速传播和不经过滤的强大力量也会被滥用，实施信息轰炸。针对这一问题，研究人员开发了工具用来鉴别垃圾信息和人们真正的心声。

Web 2.0将网络提供公平参与机会的精神发挥到了极致，从而使全球互联网进入了“社会化”时代。这是一次巨大的变革，许多Web 2.0应用，如Wikipedia、Facebook和MySpace都取得了巨大的成功。

不管是实干家还是研究人员都在考虑Web 2.0在专业机构中的用途。有人断言，Web 2.0将为企业内部和企业之间的沟通模式带来深远的变化。目前，对于“企业 2.0”的探索还处于早期。然而，已经能够预料到企业从布局Web 2.0技术中获得的收益将大大不同于以往的企业信息系统所带来的收益。

在《企业Web 2.0的指导原则》一文中，作者首先从工具、方法、结果这三个角度介绍了企业2.0中涉及的信息技术、新的工作实践以及组织目标。企业部署Web 2.0的目标主要是两点：集体创意和开放创新。为达此目标采用的Web 2.0技术主要有六个结构功能：数据和功能的反复使用和重新组合；灵活的设计、快速的更新和可适应性；内容的协同创建和修改；不会限制内容的格式；

富文本、可应答、人性化的用户界面；集体智慧的汇集。只有技术还是不够的，如何在工作中真正应用这些技术也是很关键的，要遵循以下基本原则：自主性原则、工序原则、协作原则、人员和文化原则。

自主性原则：鼓励用户主动发现技术中自己需要的用途，而不是强制其实用不必要的功能。

工序原则：让工人和管理者在经验中发现价值，在对突发模式的沟通中逐渐总结出新的工序，而不是拘泥于上级下达的教条的工序和流程。

协作原则：让人们在参与协作的过程中自发形成团体、进行协作，而不是事先指定每个人的角色、任务和行为规范。

人员和文化原则：邀请人们参与。不断地激励、引导、说服人们信服企业价值，而不是强迫他们以某种方式行事。

这些原则综合了成熟的管理理论和几个最近部署Web 2.0技术的企业案例分析，可以帮助企业从Web 2.0投资中获得最大回报。要成功为企业引进Web 2.0技术，必须摆脱过去家长式的强制性沟通策略和组织结构，转变成多对多、去中心化的自下而上的沟通结构。

过去60年的时间里，我们目睹了一次重大科技革命的开端——信息革命。信息科技带来了许多新的经济领域，并且提高了劳动生产率。信息革命还远未结束，数字技术的使用会越来越广，更多的劳动者会服务于信息经济。信息科技的快速发展引发了对学术领域中计算机与信息（C&I）研究和教育的

重新审视和重整。

《计算机与信息科学：一个学科，多个专业》一文中，作者就计算机与信息这个大学科进行了讨论。针对计算机科学究竟是属于科学、工程还是其他领域的争论，作者首先表达了自己的看法：计算机与信息是一个用途驱动的研究学科。

算法、编程语言、编译器和操作系统的早期发展所依赖的不外乎于C&I和它的数学基础。但是，如今数学已经不再是C&I的唯一基础。硬件和信息物理系统（cyber-physical system）的研究需要良好的物理学基础。不断出现的C&I

研究领域（如人机交互、社会化计算、图形和可视化以及信息检索）需要对社会科学的深刻理解（如心理学、社会学、人类学、法律学等等）。神经学科的知识可以帮助完善人工智能系统，生物学可以对未来的计算设备有所启发。C&I被这些学科所影响，也反过来影响这些学科。

像C&I这样一个用途驱动的学科，正确的组织原则不是共同的学科基础，而是对C&I系统用途的共同关注。C&I研究人员不需要成为每个领域的专家，但是有不同基础知识背景的研究人员常常需要共同协作。因此，我们

要建立一个大家能够彼此坦诚、共同合作的环境。

IT行业中，有非常多样化的工种，因此也要求C&I教育能够多样化，设置各种多样化的教学规划。但如果要将C&I当做一门学科对待，就需要为这些规划设定共同的核心。这个核心要是简洁的，而且不能妨碍之后的多样性和专业化发展。C&I已经获得了长足的发展，对我们的社会产生了深远的影响。信息技术还会继续为我们的社会带来更多变化，为了促进这种变化，我们必须努力改善C&I的学术环境。

Technology Review 2011.03



Square是Twitter的创始人之一Jack Dorsey新创立的一个项目，让人们可以利用智能手机进行信用卡支付。这项创新将极大地改变现有的交易方式。

《新货币》一文带我们了解了Square的方方面面。

2008年，Dorsey的Twitter董事长职位被另一位创始人Evan Williams取代，于是他有了开创一门新事业的念头。在旧老板Jim MaKelvey的启发下，他开始考虑将智能手机改造为信用卡支付终端。2009年12月获得了1000万美元的投资后，建立了Square公司。

许多普通商户不接受信用卡支付，其原因主要是信用卡支付系统过于复杂、不透明和价格昂贵。而Square公司提供的解决方案只需要一款称做

“Square”的外接磁性设备（通过音频插孔连接到智能手机，并且由Square免费提供），外加智能手机上的一款应用程序。只需将信用卡在“Square”设备前晃动一下即可完成支付。并且，Square公司对每笔交易收取的费用低于现有的信用卡支付系统。

Square是两种趋势催生的产物，一个是无线互联设备的不断增长，另一个是人们越来越倾向于选择卡片支付而不是现金。Square目前已经吸引了不少用户。从2010年10月份该系统正式启用到现在，共有10万个活跃用户。2011年1月份，又有65000个新注册用户。Square的潜在用户数目巨大，其中包括美国境内已经部署传统信用卡支付系统的2700万商户和还在使用现金和支票

作为支付手段的3300万商户。该公司还计划在2012年将业务扩展到美国境外。Square希望2011年处理的交易额能达到10亿美元，目前其每周的交易额已经达到了200万~1000万美元。《华尔街日报》对该公司的估值为24000万美元。

面对广大的市场前景，很多大公司也开展了移动支付业务，比如VeriFone的Payware Mobile，Intuit的GoPayment以及TF Payments的FocusPay。但Square面临的最大的威胁不是这些外部的竞争者，而是一些亟待解决的内部问题，比如设备使用上的不便、灵敏度不高的问题，与iPhone 4的兼容性以及支付上限值的限制。虽然目前Square向用户提供外接磁性读卡器，但它不是一家硬件公司。它的创新性在于创造了一种颠覆性的更

加简单的支付流程。而Square最令人感兴趣的并不是它提供的简便支付方式，而是从支付数据中挖掘出的有价值的信息，虽然这些信息的具体价值目前还无法估量，但可以肯定，它们能够帮助商户做出有力的决策，促进其成长。

这是继2010年之后，《Technology Review》第二次评选最具创新能力的50家科技企业，涵盖网络和数字媒体、计算和沟通、能源、材料、生物制药和交通运输五大类。共有21家IT企业入选，让我们来看看它们是哪些企业以及入选的原因。

计算和沟通类

苹果：其他消费电子厂商都纷纷追赶iPad的脚步。虽然iPhone的市场份额有所下滑，但仍是智能手机的标杆。

ARM Holdings：为移动设备开发能效高的定制芯片。

HTC：其Android系统智能手机挑战了Apple的霸主地位。

IBM：为许多基础设施行业提供计算服务，扩展了信息技术的市场。

IRobot：其开发的机器人可以代替人类完成危险的工作。

Calxeda：其技术可以减少数据中心的能耗。

Lyric半导体：它们开发的微处理器基于概率而非传统的二进制逻辑。

PrimeSense：为微软的Kinect设备开发了动作追踪的3D传感系统。

Square：其开发的技术，可以允许任何人使用智能手机接受信用卡支付。

网络和数字媒体类

Akamai：它的算法可以优化在线内容传递的路由。

Amazon.com：它的Kindle阅读器使得电子书最终成为巨大的主流市场。

Google：虽然该公司已经非常庞大，但它的软件开发速度相对来说还是非常迅速。在推动Android系统开发的同时，仍然致力于网络搜索。

Netflix：低廉的视频点播，通过互

联网传送的服务，低廉的线缆，为电视的未来指出了一条道路。

Cotendo：它开发的技术可以有效地对互联网流量进行路由，让网站反应更快。

Crowdcast：收集基层人员的观点，帮助企业做出更明智的决策。

Facebook：这个社交网络已经成为了强大的广告媒介以及为其提供扩展服务的创业公司的平台。

Groupon：为本地企业的促销活动提供了一种能够保证获得回报的方式。

Layar：它的开发平台使得企业和广告商都可以为自己的应用添加增强现实技术。

Twitter：其业务模式让广告商能有放矢，并且通过开放其内容给搜索引擎来获得收益。

Ushahidi：其开源众包工具可以在地图上发布现场报告，在紧急情况发生时提供性命攸关的信息。

Zynga：其社交游戏为用户提供了新的在线互动方式。

Google是否能实现成长计划取决于它是否能让整个互联网变得快捷。

《慢动作的互联网》一文指出，对Google来说，现在的互联网速度已经不能满足需求了。以Google的Chrome上网本为例，虽然它带来了很大的方便——允许用户从任意位置远程访问“云”中的数据和程序。但是目前的互联网还不足以满足Google的预想，上网本会因为网络的速度问题，而无法像本地的PC那样流畅地使用。不仅仅是上网本，在目前的电脑上，使用网络应用的时候也会遇到种种不便。这对Google来说是个大麻烦。

Google现在依赖于其在搜索引擎方面的优势开展利润丰厚的广告业务，但是它也明白其业务不能永远依赖于搜索。Google开发了很多产品来增加人们花在网络上的时间——在PC或者其他设备上——但这些产品都没

有获得很大的收益。所以Google瞄准了新的方向，基于无处不在且快速的互联网，为移动设备开发软件。所以Google的雄心壮志是：加速整个互联网，让“互联网更快捷”。

影响互联网性能的问题可能出现在任何步骤：网站服务器可能会很慢；浏览器处理代码的效率可能不高；代码可能不易于处理；几十年前制定的网络协议不符合现在在互联网应用要求的速度和交互性。但是用户对微小的延迟都会很敏感。所以Google的目标是“修复整个互联网”。

Google考虑了使互联网更快捷的各种途径，包括：改进Google自己的网站；开发JavaScript处理效率更高的Chrome浏览器；开展宣传活动，传播“最快的浏览器才是最好的浏览器”的观点，促进各个浏览器的速度改进；将网站速度纳入Pagerank排名标准，迫使各个网站改进其性能；并且于2010年底，发布了一款网站管理员工具，可以分析网站，并且自动修复造成网页缓慢的问题。

Google还将进一步深入到互联网的基础架构中去。它提出了一种新的网络协议SPDY，据称可以比现有的协议快两倍。但是虽然大家都意识到旧的网络协议导致网络缓慢，但是要替换它并非易事，这意味着要更新用户的操作系统，更换服务器、网络硬件以及网络中的其他设备。Google还计划督促互联网服务提供商提供高速的连接。但即使有了更高的连接速度，软件也需要重新设计以充分利用更大的网络容量。部署必要的基础设施也是非常费时费力且价格昂贵的，况且基础设施部署中的一些问题不是凭Google一己之力能够解决的。

最后，即使Google的互联网加速计划能够成功，但途中仍不免遇到障碍。不过这家公司非常相信自己的能力，一定能排除万难，取得胜利。

感谢译者下斌的支持。

观点



@mujiang

如果你想找一种被诅咒得相对较少的编程语言，PHP和Python看起来还是较好的选择，<http://bit.ly/hOhhiU>。



@powerwu

将要见证淘宝和阿里云在PB级别分布式计算上的PK，是开源传万世，还是自主知识产权获胜呢？年底见分晓。



@yining

@Fenng：一个新的技术Buzzword正在逐渐热起来：#DevOps” // 搭推推荐一个Blog：agileweboperations.com。另外据非常有限的了解，#anjuke的#Devops做得很不错。



@sagacity

就目前的趋势看，高度可伸缩系统架构和用户体验设计，一个后端一个前端，会成为热门技能、职业和主题。相关技术也会走向普及，但是相信不会满大街都是……



@huandu

创新工场的行云看起来非常Promising，不知道实际如何。从官方介绍来看，好像已经把各个社交平台接入工作，以及基本的游戏B/S通信和页面工作都做好了提供出来，如果真的如此，那确实很有难度。



@mikespook

6款用于在线游戏的基于HTML5的JavaScript游戏开发框架，<http://url.cn/12mq13>。

资源



@bluedavy

趋势科技中国研发中心SPN研发团队的Blog，非常不错，对于研究Hadoop/HBase的人尤其值得仔细看看，<http://t.co/oxqXHML>。



@panweizeng

分享我的工作环境配置，包括Vim/Bash/Screen等，有时间还会将~/bin下的脚本整理进来，<https://github.com/panweizeng/env>。



@xu_lele

<http://slidesha.re/e03jla>, RT 今天的又一份重量级Slides。用MongoDB+Node.js搭建一个Social Game的成功案例。来自Ameba的“屁股（ビグ）”团队。用作者的话说是爽得连鼻血都出来了 #mongotokyo。



@greghe728

Toucheegg让Linux也支持多点触摸手势，<http://bit.ly/fsGJMc>。

□ ————— 文章 ————— □



@shizhao

漫话中文分词算法, <http://goo.gl/bfusx>。



@Beichen

如何仅用300行代码养活自己一年, 并将公司卖出?
<http://goo.gl/PsQue>。



@ideawu

PHP查询MySQL大量数据的内存占用源码分析和优化,
欢迎大家捧场! <http://www.ideawu.net/blog/archives/581.html>。



@gowers

想开发一个自己的操作系统吗? 相信我, 依次读完后你
也行的! <http://goo.gl/ZWEoz>。



@lifesinger

JavaScript 禁用与屏幕阅读器, <http://wp.me/p9YN-1R>。



@ecvip

王兴的创业观和知乎的价值点, <http://www.geekpark.net/event/view/6>。



@zheng

创业工具: <http://knb.im/7yl>, 收集整理的蛮全, 适合小
创业团队使用。



@lidaobing

Django 1.3进入RC阶段, 考虑到Rails和Django经常互
相学习, 用Rails的人也可以看看, <http://goo.gl/QxEC6>。

(网址链接请访问杂志官网, www.programmer.com.cn)

CSDN最受关注文章 TOP10

(2011年2月26日~3月26日)

1. 2011程序员薪资调查揭晓: 5年和5000元是分水岭
2. 智能手机的重新洗牌: 三国鼎立
3. 淘宝全面开放, 三年投三亿不和开发者争利
4. 第一财经周刊: 诺基亚之死
5. 华为就3Leaf事件发公开信: 邀美国当局对其调查
6. 腾讯小小警告Groupon: 中国地盘听我的
7. iPad 2真机试用图赏
8. Leslie Valiant获得2010年图灵奖
9. MSN Spaces 3月17日正式关闭
10. Foursquare的“谷歌时刻”

JavaEye最受关注文章 TOP10

(2011年2月26日~3月26日)

1. Web开发人员应当知道的15个开源项目
2. 开发人员、程序员与计算机科学家三者之间的区别
3. 在西方的程序员眼里, 东方的程序员是什么样的
4. 自由职业者创建的12家超酷公司
5. 那些员工很少却有数千万用户的互联网公司
6. FreeBSD 8.2发布
7. SimpleJDBC 1.0发布
8. Ubuntu 11.04 Alpha 3版本发布
9. 当女软件工程师的那些好处
10. Google今年打算至少收购50家公司

成功实施CRM的关键因素



Adrian Johnston

微软商务解决方案事业部亚太区总经理

微软的CRM产品自2004年进入中国以来，已有七年历史。我们发现，中国用户在对待CRM时，主要有两大误区：

首先，没有意识到CRM的成功实施，不仅取决于产品技术，尤其和企业业务流程的联系非常密切。作为企业的负责人，是否有进行真正的业务变革的决心？是否能真正地引入科学的管理流程，并把变革的过程落地？这往往是CRM项目

实施成败的关键。

其次，没有意识到CRM的实施其实也是一个管理的过程。即CRM部署的每一个阶段都要有非常好的管理流程。有时我们也很尴尬，很多的用户应用做得挺好，但最终在实施的过程中因产生了一些瑕疵而功亏一篑。或者说他本身有坚强的变革决心，但没有统筹整体的解决方案，通常都小步快走，而规模一旦扩大时往往蛇吞不了大象，最终导致管理目标不能实现。

在摒除了以上两大误区后，用户在市场上寻找优秀的CRM解决方案，一般需要考虑如下几个因素：首先是使用方便、部署容易，具有非常高的易用性。其次，确保使用它之后可以促进公司业务的效率大大提升。另外，系统要具有良好的可拓展性，不仅可容易地采纳新系统，而且可以把解决方案不断地拓展到重要的市场合作伙伴们的系统里。

以上因素，再加上价格优势正是微软CRM长期以来能提供差异化的、与众不同的产品的奥妙所在。因此微软在CRM行业中一直处于世界领先地位，目前在全球范围内有2.4万家公司用户，140万个人用户。而微软最近发布的新产品Microsoft Dynamics CRM 2011，更是沿着上述优势和脉络，继续大步前行。

第一，使用 Microsoft Dynamics CRM 2011可以提高用户的市场营销工作能力，使投资回报率最大化。现在大约98%的市场营销人员都认为，他们的预算非常紧张。因此，在CRM市场营销中，我们需要把有限的预算更加优

化地加以使用。而现在Microsoft Dynamics CRM 2011可以提供端到端解决方案，从发现线索到直接交易获得现金收益，进行全过程的管理，并保证公司工作的协调统一。同时我们还可以看到它能在整个系统运行的过程中和微软其他系统进行有效的整合，比如SharePoint等。

第二，Microsoft Dynamics CRM 2011能对客户服务的整个生命周期进行全面的的管理。现在很多公司都把留住客户作为发展战略之一，但他们往往对客户的服务并不是很到位，无论是呼叫中心，还是面对面访谈都做得不太理想。所以如果能把客户服务提升到一个新的等级的话，就会占据很多优势，并拥有很多回头客。通过CRM 2011，不仅可以快速响应客户的要求，而且我们还提供了多种客户关系的管理，无论是公司内部还是外部，无论是一般的公民服务还是医院病例服务等，都能多途径、全方位地对客户进行有效的服务。

第三，Microsoft Dynamics CRM 2011还有一个重要的特点，使得我们很多用户非常感兴趣，就是它可以帮助你完成从其他CRM产品到微软CRM的转换，可以帮助你根据自己的需求进行个性化、定制化的修改，以提高你的工作效率，并给你一个全新的用户体验。

CRM2011版本中总共增加了500多种新功能，已深受合作伙伴们的青睐，如HCL大中华区总裁顾旋表示，

“Microsoft Dynamics CRM 能够从首次接触客户起，在整个购买和售后流程中创建并维护明晰的客户数据，形成全客户生命周期的管理。使得我们可以帮助客户进行不同业务平台的无缝整合，带来更多业务提升机会。”

目前，Microsoft Dynamics已成为全球五大企业软件供应商之一。中国是Microsoft Dynamics非常重要的市场，微软会继续加大在中国市场的投资，并已制定了非常广泛的市场策略，并且首先是在纵向市场上进行一些拓展。展望未来，社交型的CRM产品和服务将成为主流，很多公司都把这种社交型的CRM解决方案放在自己总体解决方案之中。微软也对此做了很多的考虑，比如我们和Twitter就有这样的合作，帮助他们进行CRM社交的管理。微软确实有这样的能力，并且也正在计划，让未来社交型CRM的管理能力能得到大幅度的提升。P

震痛·震恫·震动

安全

二月，世界的焦点是日本，日本的焦点是福岛。在巨大的人员伤亡面前，IT乃至更窄带的信息安全都是浮云。

因此，奇虎360和网秦相继两家安全厂商走到了美国上市的关口的消息，已经不是很突出。但就在网秦发布招股说明书的前一天，遭到了3.15晚会的点名曝光。对产业来说，这也是一次“人工地震”。

对相关问题，厂商和媒体自然有不同的说法。手机应用界与传统安全界其实一直存在着某种价值观的对立。SP链条上的大小厂商认为一些扣费问题就是手机行业的基础生存模式，不过是常态而已；而很多传统的安全厂商和团队，则似乎一直没有把类似网秦这样的新兴厂商真正看作自己的同行。

有趣的是，在360崛起的前期，安全阵营也是这样看待360的。但从资本运作的角度，这一次，他们坐在了产业的前排。

当每一个中国企业走入关键的资本的关口时，我们祝福它们顺风顺水。但我们也希望，上市后的透明和监管，以及创始人向公众企业家的转型，能让相关企业不仅会延续其进取的基因，也承担起更多的产业责任。

三月，Google继续在安全发力，收购了安全团队Zynamics。这是一个小众公司，但其研发的BinDiff等工具，都是漏洞分析者的标配工具。

但Google的Android Market模式却遭遇后院起火，不得不宣布撤下了58

款蓄意带有提权漏洞代码的软件，这些软件来自几个ID的提交，基本上并非自行编写，而是对其他软件的修改和重新打包。

开源主义者一直坚持的“开源是安全的朋友”的巨大光圈基本上到了最终幻灭的程度，现在是讨论“开源是不是安全的敌人”的时候了。

而之后Google的行为也略显激进，它推出一款安全查杀工具，远程安装到已受害用户的手机上清除恶意软件。

这一举动再次引发争议，有安全团队指责如果此前被植入代码已经生效，二次下载或者注入了其他程序，那么Google的这一行为不啻于破坏现场，也无法完全消除威胁。

戏剧性的是，几天之后，又有攻击者将恶意代码植入这一安全查杀工具并广泛分发。而更让我们觉得需要关注的是，Google已经开始逐渐不自觉地走向“大神”角色，但正神也好，邪神也罢，终极都注定是一个平等而安全的信息世界的敌人，无节制的义务期待，将导致无约束的权利聚合。

国家信息安全漏洞共享平台（CNVD）发布了最新一期漏洞公报，本期共收集、整理信息安全漏洞134个，其中高危漏洞42个、中危漏洞18个、低危漏洞74个。

值得一提的是，其中包含北京某公司的工业控制软件kingview的漏洞，这让人在震网蠕虫后再次关注工业系统的安全。P



主持人：

肖新光，网名江海客，安天实验室首席技术架构师，研究方向为反病毒和计算机犯罪取证等。

本月热点

新闻

● 水军，美军的最新兵种

根据英国卫报的披露，美国军方正在开发一种软件，该软件支持一个人管理10个具有令人信服的背景和历史的虚拟身份，用于在Facebook和Twitter等网站进行活动，并且建立了一套机制以避免操作者的实际IP被追踪到。其指定的活动语言包括阿拉伯语、波斯语、乌尔都语和普什图语等。City University of New York的教授Jeff Jarvis对此的评价是“华盛顿显示出一个拙劣的垃圾邮件发送者的道德”。

● Pwn2Own黑客大赛

为期三天的Pwn2Own年度黑客大赛三月在温哥华举行。第一天，法国安全公司VUPEN利用一个0day漏洞攻破Max OS X上的Safari，另一位安全人员在Windows 7 SP1上攻陷IE8。第二天，iPhone 4和黑莓Torch 9800上的浏览器也相继被挑落。最终Firefox、Chrome浏览器和Android、Windows Phone 7系统保持坚挺，原定的攻击演示因为入侵方法不稳定而被安全研究人员放弃。



主持人:

王翔, 软件架构师, 主要研究方向为XML、.NET、领域设计和PKI应用。工作之余喜爱旅游、写作和烹饪。

新闻关键词——Oracle

本月热点

新品发布

● Devart发布LINQ连接器

Devart发布了全面改进的LINQ连接器, 提供对SQL Server、Oracle、MySQL、PostgreSQL和SQLite的多项扩展功能。

● Speedware发布企业级开源COBOL框架

Speedware获得COBOL-IT授权在北美市场发布企业级开源COBOL框架, 尽管COBOL已经远离公众视野, 但COBOL程序依然在承载着众多的“关键数据应用”, 开源或为经典注入新生。

● Schooner发布MySQL (企业版) 闪存优化技术

Schooner科技宣布, 将提供面向戴尔、惠普和IBM x86等服务器的MySQL (企业版) 闪存优化技术。

● Attunity与微软达成协议

近期, Attunity与微软达成协议, 将在下一代SQL Server中提供对异构数据库的动态变更数据抓取功能(CDC)。

技术会议

● Idera在线讲座揭示SQL Server存储过程执行内在机制。

数据库

2月19日, 甲骨文首席执行官拉里·埃里森出现在一个技术领袖的高峰宴会上, 尽管媒体更多将焦点放在乔布斯身上, 但本月的技术新闻中甲骨文及周边厂商围绕“Oracle”的各种消息一直颇为暖场。

甲骨文一直将其数据库定位于企业级应用, 那么“企业级”信息系统时的技术热点是什么呢? 答案自然挂一漏万。但通过虚拟化实现宏观数据中心管理、高可用、高并行处理、商务智能、SOA化梳理IT系统、信息安全一般都会出现。而甲骨文的技术新闻中除了关于高可用、商务智能、数据中心的常规“不新闻”的新闻外, 其他热点也在本月频频亮相。

为了向大型客户提供更加“按需”的存储和数据支持, 甲骨文发布了云文件系统, 帮助用户迁移到私有云, 更简便实现数据存储管理。新的文件系统充分发挥了甲骨文在集群和备份、恢复方面的固有技术优势, 将以往众多Oracle数据库上系统管理能力延伸到存储系统, 是甲骨文近几年数据库管理、虚拟化以及与亚马逊合作经验的一个汇总。

为了满足大容量历史输出的存储要求, 甲骨文推出了StorageTek T10000C带库产品, 通过采用分层存储为大型数据中心提供单位成本更低的解决方案, 对于那些今年背着硬性节能减排指标的机构, 采用大容量带库也不失为一个选择。该产品速度是亮点, 具有5TB的固有容量和240MB/秒固有吞吐量, 而且可以在线加密, 可

简化机构历史数据安全管理要求。

信息安全方面, 甲骨文发布了被其称为“数据库第一道防线”的Oracle Database Firewall。通过新的语法分析引擎, 外部提交的SQL语句可以根据设置的策略选择通过、计入日志、报警、阻断或是被替换, 通过该措施可以将很多SQL注入攻击屏蔽掉, 尤其随着开发工具和语言的发展, 采用老办法对付新SQL注入手段往往只能亡羊补牢, 而通过数据库防火墙则可以做“先手”控制, 也就是“你有千条妙计, 我有一定之规”。对于我国“走出去”到美国的国际化企业而言, 通过数据库防火墙预置的报告模板, 可以简化一些隐私和管制法令所要求输出的报告。

对从事Java EE的开发团队, 甲骨文新发布的GlassFish Server是不错的开发环境应用服务器或生产环境临时用品, 通过JPA或JDBC开发团队可以更方便的连接Oracle, 获得更接近于WebLogic Server的功能体验。通过新版GlassFish Server内置的OSGi支持, 开发团队可以直接与众多移动终端、智能设备、物联网设备交互, 而以往类似的中间件产品价格不菲。对于同时连接多个分布式部署的Oracle数据库环境, 可以配合Oracle Access Manager实现单点登录。

凭借对于NoSQL的包容、收购再孵化等合力, 这个春天甲骨文围绕数据提供的一系列技术迎来了收获季, 所有这些令业界对未到的那个也许叫Oracle 12g的主角充满期待。P

开放平台， 企业应用开发新蓝海

企业应用与行业软件

仅针对程序员领域，2011年如果问什么名词最火的话，它既不是移动互联网，也不是云计算，更不是电子商务，它的名字叫开放平台。

2月23日，2011淘宝开放年战略发布会在北京举行，直接推动了开放平台的井喷。我觉得，2011年才算得上是开放平台的元年。事实上，《程序员》杂志也在上期专门做了《2011开放平台之征》的专题。截止2010年底，淘宝开放平台上已拥有11万注册开发伙伴，每天淘宝API调用数达7亿次之巨。同时，到2010年淘宝已经衍生出围绕电子商务的多个种类服务行业，目前淘宝平台上通过淘宝网认证的合作伙伴达到200家，涵盖运营、客服、营销、仓储、培训等行业。

碰巧的是，去年年底我恰好写了一本图书《B2B2C开发指南——基于SaaS和淘宝API开放平台》，也将于3月底正式出版发行，大家看到这期杂志的时候，也是我这本图书发行的时候。看来大家所见相同，开放平台真的是一块“大金矿”。

如何挖掘这块“大金矿”？我在《程序员》杂志2011年第2期，写了一篇《开放API，让企业管理软件连起来》的文章，讲述了我们用互联网的基因，来解决企业应用开发领域的关键问题：垂直化（专注某一领域）、标准化（做产品）、产业化（形成上下游产业链）、规模化（面向草根而不是大客户）。而像淘宝、新浪微博、开心网、人人网等用真金

白银砸出来的流量，开放了自己的平台API，为小公司和草根创业者实现“四个现代化”，变绝对不可能转化为无限可能。没有开放平台，小公司和草根想自己做，几乎是不可能。

开放平台还不算尽善尽美。淘宝的API几年内数度易稿，甚至连类名、方法名、参数、包名都大规模修改过。我调试某微博API，无论如何都通不过，上官方论坛，发现整整一周时间，都有人抱怨同类问题，而该微博非正式宣告API暂时休假了，一周后再重新开发，很不重视平台上开发者的利益。在开放策略上，也都是半抱琵琶半遮面，每家都会写上，“最终解释权归本公司所有的”霸王条款和潜规则，开发者还处于弱势群体，终究没有办法。

盛大集团董事长陈天桥在今年两会上也提到了类似问题。“中国现在有18个号称开放的互联网企业，但是，有多少企业在以开放之名，行封闭之事？试想一下，如果移动的号码不准打联通的号码，那联通会死掉的！移动领域的互联互通，大家可以接受，怎么到了互联网领域就不行！”

虽然我们不得不忍受的潜规则太多，但正如上文所述，开放平台，为我们提供了无限可能，利远远大于弊，套用某一位想一夜成名的女演员的一句话：“只要有潜规则就好”，还是让我们愉快的投入开放平台怀抱吧。P

主持人：

邢波涛，现任北京新软孚信息技术有限公司技术负责人。关注SaaS管理软件和B2B、B2C电子商务的融合。



本月热点

事件

● 诺基亚出售 QT 业务

Engadget报道，诺基亚决定将QT商业授权和服务业务出售给Digia公司，Digia计划雇用19%的诺基亚技术咨询服务团队和全球Qt商业销售和营销团队。诺基亚还将转交大约3500家使用QT的桌面和嵌入式客户。Digia表示会继续投资开发Qt。看来史蒂芬·艾洛普真的是微软派到诺基亚的特洛伊木马。

● 西部数据43亿美元收购日立硬盘业务

WD西部数据公司3月8日在全球硬盘市场投下一颗重磅炸弹！这家全球最大硬盘厂商宣布，将以现金加股票的形式，出资43亿美元收购日立全资子公司。

会议

● 2011淘宝开放年战略发布会

2月23日，2011淘宝开放年战略发布会在北京成功举行，至此，2011淘宝开放年正式拉开帷幕！



主持人：

姚磊，Microsoft Dynamics CRM MCP，多家企业信息化商业解决方案项目经验。熟悉IT规划与需求工程与项目管理。

Social CRM

本月热点

新品发布

● Oracle发布了Java移动开发框架

经过两年的研发，甲骨文发布了一个移动客户端和相关框架，以帮助开发人员快速构建工业移动设备的Java应用程序。Oracle应用开发框架（ADF）移动客户端应用程序开发提供了可以在不同的移动平台部署而不必要为每个特定平台编写代码。

事件

● 传Oracle再掀收购 以78亿美元收购AMD

甲骨文在硬件市场频频发力，2009年以74亿美元鲸吞Sun，如今又将AMD收入囊中，凭借这两起惊天并购，甲骨文同时拥有了RISC和X86两条CPU产品线，完成了在芯片产业的布局，同时也将极大提高自身和IBM在服务器市场正面对抗的能力。

会议

● 中国云计算服务大会在沪召开

为期两天的中国云计算服务大会于2011年3月16日在上海召开，此次大会得到了国家工信部、科技部、上海市政府和上海经济信息化委员会的大力支持。大会吸引了近300位业内专家，其中包括了政府官员、信息化专业研究机构与专家学者、全球知名企业首席信息官、IT信息经理，以及全球领先的云计算技术服务提供商等。

推荐资源

● 做高效人士必看：《高效人士的7个习惯》

作者：斯蒂芬·科维（Stephen R.Covey）

企业应用与行业软件

未来5年软件行业主旋律在于“社交型CRM”，将SNS功能融入CRM中，利用Social CRM系统为企业建立并维护客户关系，将成为企业发展的新思路。准确把握社交网络的脉搏可以促进变革，提高客户忠诚度，并刺激销售，提高服务质量。

社交网络与我们现在常谈及的CRM有一定的相关性：

关于沟通：随着社会化媒体的深入，信息的获取途径也在发生变化：CNN在一项调查中发现，43%的新闻通过社交媒体网络和工具分享（如Facebook、Twitter、YouTube和MySpace等），紧随其后的是电子邮件（30%）、短信（15%）和即时消息（12%）。

关系价值：人所具有的网络圈子关系，在很大程度上决定了人的社会价值；“关系”将是可网络交易的商品；关系总是和隐私、秘密和情感等主观因素紧密联系，随着网络中（关系）的应用，整个世界的扁平化和流动性都在大大增强。

交易成本：社交网络建立的是关系推荐引擎。信息推荐使商业广告的效果更容易衡量，关系推荐使人与人之间交往的商业效果变得更容易衡量。在关系交易过程中，各种联系人都有可能获益，这样就把信任关系的效率最大化。

各大厂商都在社交CRM上做出了反应，甲骨文公司推出了销售人员的Oracle CRM网络小工具，把企业CRM系统的公司数据和互联网的公用数据

进行整合，让销售部门更好地获取用户数据，实现无缝管理。该产品的其中一个特性就是将来自内部系统的销售订单数据与外部信息混合在一起，来帮助预测销售机会；另外一种特性则是模仿Facebook网站的功能，使销售人员可以创建和加入群。

以在线CRM起家，并致力于云计算发展和应用的Salesforce公司针对企业用户推出名为“Chatter”的社交网络平台，以方便企业用户的员工共享各类信息，并加强工作协调能力。借助该平台，企业员工可上传个人资料、实时信息和状态更新，同时把应用程序和数据整合到Salesforce平台当中。其他如Google、Zoho等在线软件提供者，同样也在进行软件与SNS网站，如Facebook、Myspace等的融合。

不同于国外产品强调与SNS的融合，本土软件企业还纳入了企业内部的协同管理，将传统的层级式、流程化的协同管理转化成社交网络式的管理，大大增加了传统的协同办公的灵活性和效率，让管理更加人性化。这种方式并不会颠覆传统协同软件在流程管理、权限管理等方面的地位，但作为一种补充，SNS与在线CRM的融合会给协同管理开启新的大门。

从技术架构搭建的合理性来看，社交CRM考虑得更加人性化，充分考虑到使用环境的特殊性，结合瘦客户端和云计算等多种方式。

另对企业行业人际关系价值的评估体系和价值链关系网应用的构建有待更深入探讨。P

反射在游戏开发中的应用

游戏开发

上一期专栏，我曾提到.NET提供了一个特性叫做反射(Reflection)，其实反射并不是.NET所独有的。在这一期专栏，我想和大家讨论一下反射这个特性，在游戏开发中的应用，以及发展前景。

在原生的C++程序中，我们通常会定义一些类，并且为这些类写一些方法、成员等。这些方法和成员，实现了某个游戏功能，或者记录某些数据。通常它的作用区域都是存在于编译器编译期。通过C++编译器，这些类、函数、变量就会变成一些实实在在的地址，执行模块也会给这些取一些只有它自己才知道意义的名字。然而，我们有时候会有这样的想法：我能不能在某种时候，调用进程里名叫“GetNPCName()”的函数，并且，还能给它传递所必须的参数，以得到这个函数的返回值呢？

显然，在原生C/C++一类的静态语言，目前是不具备这样能力的，而Python一类的解释性脚本，显然要做到这个要费一些周折，并且性能十分低下。在.NET框架下，引入了一个数据结构描述数据——元数据，在元数据里面，它记录了编译器编译过程中获得的各种信息，包括类名、函数名、函数参数、函数返回值、变量名、变量相对偏移，等等。在应用程序集加载的时候，这个元数据被加载起来，当用户需要这些数据，可以通过一些特殊接口来获得，并且处理执行。

那么拥有Reflection这个特性后，我们在游戏开发中，可以获得哪些好处

呢？这里简单介绍一些常用的应用：

- 编辑器数据和UI的绑定。有了Reflection特性，我们可以做一个统一的规范，让编辑器界面和程序内数据结构直接映射，减少大量繁杂的关联代码。

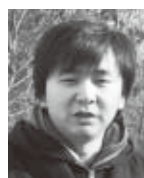
- 游戏对象的动态创建和初始化。在MFC一类的类库中，通常会提供一套RTTI（运行时类型信息）来做动态创建，创建和初始化参数等处理起来就不会力不从心了。

- 游戏内功能模块组合配置化。比如要做一个流程处理，某种情况下调用某些代码工作。但是希望这个调用能灵活配置，那么可以通过一个配置文件，或者数据库字段，获得处理函数入口和参数。这样，后面的各种功能配置组合，都可以交给策划去填写配置文件或者数据库字段。

- 类似游戏内图文排版一类需要高度灵活可配置的模块，用Reflection是一个很不错的选择，可以比较轻松面对各种调整修改。

在.NET平台，配合Reflection，还有Attribute系统。Attribute的作用，是对用户所写的代码进行一些标注。比如可以给一个类添加一个Attribute，打上一个标注，这个类是用来描述NPC行为的，然后再给这个类成员函数标志Attribute，申明这个函数是用来处理NPC被交互的时候调用的，等等类似。

然后可以自己写程序，搜集程序集内，所有打上这些标记的类和函数，然后统一做逻辑处理。P



主持人：

宋忆疆，参与《碧血情天》，《傲世三国2》，《乱舞天下》等游戏研发。目前担任《流星蝴蝶剑OL》项目制作人。

本月热点

新品发布

● 金山公布Q版回合制新游《大话红楼》

3月7日，金山正式公布2011年2D Q版MMORPG玄幻回合制新游《大话红楼》，本游戏改编于中国四大名著《红楼梦》。官方工作人员透露将于本月邀请999名回合制专业玩家对游戏进行内部封闭测试。

● 《流星蝴蝶剑OL》二测开启次时代动作网游风暴

2月25日，久游MMOARPG《流星蝴蝶剑OL》二测面对玩家开启，游戏开放了6个职业，前40级内容，大量玩家进入体验了真正的国产次时代动作类武侠游戏的魅力。完美《笑傲江湖OL》和蜗牛《九阴真经OL》也将很快面对玩家进行首测。

● 暴雪COO称Titan已可试玩，将超越魔兽世界

3月，暴雪COO Paul Sams在接受媒体采访时透露了暴雪新MMORPG（代号：Titan）的最新情况，他认为Titan是一款非常棒的游戏，相信它将会超越魔兽世界。

主持人：

钱宏武，现任职盛大创新院，原搜狐互动产品开发部主管，资深互联网社区架构师，垂直搜索领域专家。



3月的浏览器

本月热点

新品发布

● 微软有望凭IE9阻击Chrome和Firefox

尽管微软的IE仍在浏览器市场上占据了大量的份额，但目前这一产品已经面临着Firefox、Chrome、Safari以及其他竞争对手产品的巨大竞争压力。IE9正式面世后，如果能够获得消费者的大量青睐以及企业用户的抬举，必将有利于微软遏制或扭转其IE系列产品在浏览器市场上份额缓慢下滑的局面，但未来会如何，微软必须学会跟上互联网的脚步。

● Google发布针对IE 9的WebM插件

IE9正式版尽管已经提供了对视频的原生支持，但是显然只有H.264编码的份儿，Google自家的WebM就得靠边站了。既然微软给Chrome准备了H.264扩展，那Google何不也倒打一耙给IE9提供WebM插件呢？从来就没有什么救世主，但分清轻重很重要。

Web

年初都是浏览器集中更新的时候，浏览器将扮演越来越重要的角色。这是拉开未来大幕的重要预示。

与微软以往经常跳票不同，IE9这次在3月15日按时发布，很多媒体都对这款新浏览器好评有加，认为微软有望凭借IE9阻击Chrome、Firefox等浏览器的市场份额增长。

让我们看看IE9有哪些变化。首先是界面更加简洁，只有一行控制区域，包括功能合二为一的地址、搜索栏、标签页和前进、后退等功能按钮，可谓大道至简。

此外，还有几个功能变化，最值得注意的是对HTML5的支持（并非所有HTML5特性），已经可以基于此来开发出很炫的游戏了。

浏览器是未来游戏的重要平台，HTML5技术的重要性在于真正的跨平台，休闲游戏已经有很大的用户基础，随着手持设备种类越来越多，完美运行在各个平台只有依靠HTML5技术。只要基础界面合适，写出的程序就可以运行在所有平台上，这不光降低成本，还能将不同设备的用户整合到一块儿，产生巨大的效益。这一切都需要HTML5，而曾被诟病保守的微软都借助IE9跨出了这一步，我们有什么理由不去拥抱变化。

作为IE9的大敌，Chrome的发布时间和功能的针对性，都显示出Google巨大的策划能力：Chrome 10中添加了V8引擎Crankshaft，这是自2008年首次发布Chrome以来最大一次性能提升。新的引擎使得Gmail等Web程序启动速

度进一步提升，富JavaScript代码网页加载速度提升了20%左右。Windows7平台JavaScript基准测试结果显示，Chrome 10的10项测试（平均）积分为728，IE9平均积分仅为465。

Google还帮助微软完善IE9，发布针对IE9的WebM插件。这是新时代的活雷锋吗？非也，因为微软也为Chrome准备了H.264扩展。行业两巨头思路极其清晰，一时义愤是没用的，控制标准才是王道，浏览器只是推广标准的渠道。渠道和王道，孰轻孰重，这帮最聪明的人不可能搞错。

Firefox肯定不会错过升级的热闹，Mozilla基金会三月底发布Firefox 4正式版，其中的新功能包括名为JetPack的新插件构架、精致的用户界面以及改进性能的JavaScript引擎。此外还包含名为“Panorama”的新功能，给用户提供更全新的浏览器标签管理方法，主要特点就是对标签分组管理，在一个页面上显示所有打开的标签，以及在浏览器标签栏只显示选择的标签等功能。与此同时，在Firefox 4中，用户还能够在多个设备之间同步设置、密码、书签、历史、打开的标签和其他自定义。此外，同IE9一样，Firefox 4也具有“禁止跟踪”选项。

虽然每个浏览器升级的功能可能都不同，但JavaScript受到了同样的重视，这个当年看起来一个不起眼（甚至还被认错）的脚本语言，今天成为所有浏览器的焦点。这个就是互联网，你永远不知道未来谁会是耀眼的明星。P

智能平台安全之战

移动

第一季度，各Pad开发平台都发布了新产品。3月11日，iPad 2正式上市，没有预期的更高分辨率屏幕，也没有Retina，但双核A5处理器、更多内存、双摄像头、陀螺仪，更轻薄的机身、同样优异的待机时间、持平的定价，还是给苹果爱好者带来了惊喜。Jobs在iPad 2发布会上大打价格牌，嘲笑高价的Xoom。Xoom的问题不仅是高昂的售价，缺乏应用、不够稳定以及糟糕的电池续航能力才是致命的。前两者会逐步改善，后者则需要软硬件开发方一起解决，其根由是Android系统的无限后台多任务系统的设计导致的，这个架构问题，并非简简单单就能解决。

Google第一季度接连发布了多个Android新版本，其中2.3.3带来了新的NFC和Bluetooth相关的API，SDK版本也升级到了10，代号依然是Honeycomb。随后Google发布了Nexus One和Nexus S的2.3.3的OTA升级包，对于2.3.0至2.3.2版本，类似于2.0和2.0.1，属于不稳定时期的过渡版本。此外，Web端Market新版本的推出给用户带来很大的方便，可直接在PC上安装和升级应用。随着iPad 2的发布，苹果推出了新的iOS版本4.3，主要变化是Safari上的JS引擎性能增强以及增强的Hotspot功能。对苹果开发者来说，新闻是Xcode新版本4.0开始收费了，但只有4.99美元。WebOS推出了新版本2.1，取消了对Flash的支持，增加了包括中文在内的多语言。Nokia在3月11日宣布正式加入微软的

Windows Phone 7阵营，Nokia首席执行官Stephen Elop解释说，此举是为了避免使移动领域发展成为由Android和苹果iOS两家垄断的市场。两个正在衰落的巨人，能否结伴走出困境，还要等待市场的验证。

智能手机平台上连续暴露了几件安全问题，影响最严重的是Android的木马门事件，不良开发者在Android Market上发布了几十款破解过的应用，并嵌入木马。利用Android 2.2.1及之前版本的漏洞获得root权限，安装了一个特殊应用，可以上传用户任何资料，在用户不知情的情况下偷偷给手机内安装新的应用，而且恢复出厂设置都不能清除。iOS上同样也不安全，黑客可以通过特殊设计的网页使Safari中毒，上传用户数据。其他平台上，WebOS/WP7，甚至一直打安全牌的BlackBerry同样暴露过一些安全漏洞。智能手机就是小型PC，无论是否开源，安全问题必然存在。

Android Market对应用的不加审核，导致了不良应用在Android平台上更易于传播，对此Google有着不可推卸的责任。国内，在有无数第三方缺乏管理的应用商店以及下载渠道的情况下，使得此问题更加严重。破解第三方应用，植入自己的广告和吸费代码，甚至植入木马，监控及上传用户数据。由于大量越狱的iOS设备存在，在iPhone和iPad上一样存在这种问题。无论如何，可以想象在未来几年内，手机安全方面的应用一定会越来越热。P



主持人：

崔海斌，十几年的开发经验，专注于Java、Android、移动开发等领域。现于创新工场旗下的点心团队就任总架构师。

本月热点

事件

● Android平台首次超越黑莓

3月8日，市场研究机构comScore发布的最新数据显示，截至2011年第一季度，Android平台在美国市场的份额达31.2%，首次超越黑莓成为美国最大的智能手机平台。在这3个月中，美国智能手机用户人数达6580万人，环比增长8%。其中，31.2%用户使用Android手机，环比增长7.7%；30.4%用户使用黑莓手机，环比下滑5.4%；24.7%用户使用iPhone，环比增长0.1%。

● 民间组织对Google提出了7点要求

一个民间组织Android Developers Union（Android 开发者联盟）成立，对Google提出了7点要求。此7点要求其实都是针对Google的Android Market提出的，主要围绕税费、支付渠道、沟通方式和排名算法等方面。不过我自己还是觉得开发环境问题也是非常突出的，尤其是模拟器的性能低下，极大影响了开发者的开发效率。

主持人:

刘洋洋 (花名额台), 淘宝网前端开发工程师, 负责淘江湖及相关产品的前端开发工作。



前端开发流程自动化

本月热点

新品发布

● IE9推出

3月15日, 微软在影视音乐互动大会上宣布推出最新版本的Windows Internet Explorer 9浏览器, 支持39个语言版本, 不支持XP系统。微软IE9从测试阶段就受到了业内外的极大关注, 性能、标准化等都较旧版本的IE有了较大提升。IE9发布后24小时内, 下载量高达235万, 希望能对国内混乱的浏览器现状带来积极的影响。

● iPad 2发布

3月3日, 苹果发布iPad 2, 同时推出iOS4.3, iMovie、GarageBand等重量级应用也登陆iOS。同日, iPad一代大幅降价, 新款iPad于3月11日在美国开售。2010年苹果iPad销售1500万部, 占市场份额的90%。同时App Store的软件数量和下载量也不断飙升, iOS开发已经成为当下最流行的话题。

新闻

● MSN Space关闭

3月17日, 微软Windows Live团队发布公告, 宣布Windows Live Space服务正式关闭。至此, 曾拥有3000万用户、经营长达7年之久的MSN Space正式退出历史舞台。

Web前端

如今前端工程师需要维护的代码变得极为庞大和复杂, 代码维护、打包、发布等流程上浪费的时间精力也越来越多。精简流程、提高效率, 是每一个前端团队都会遇到的问题。大部分前端团队使用Ant脚本进行这一系列流程的自动操作。

使用Ant

Ant主要用于代码构建、打包、部署的自动化操作。早先主要用于Java开发, 但由于它具有接口开放、便于配置、Java跨平台等特性, 在前端流程自动化方面同样可以发挥强大的功用。

代码检验: Ant + JSLint

JSLint是最常用的JavaScript代码检验工具, 它使用JS实现。Ant提供了scriptdef标签, 可以直接解析JS脚本, 但需要对JSLint代码进行一些修改, 以便于控制台输出。也可以使用jslint4java等开源程序, 或通过Rhino等JS解析器方便地进行自动化代码检验。

文档生成: Ant + JsDoc Toolkit

JsDoc Toolkit是基于JSDoc, 使用JavaScript开发的JS文档生成工具。开源工具JsDoc Toolkit Ant Task对JsDoc Toolkit进行了简单封装, 可供Ant调用生成文档。

打包合并: Ant

文件合并可以使用Ant内置的concat标签实现。代码压缩可用Ant + YUI Compressor实现。

测试环境部署

使用Ant原生的copy标签, 可以将本地代码复制到局域网内的ftp目录。也可使用开源工具JSch进行SCP操作。

SVN提交: Ant + SvnAnt

SvnAnt是一个开源工具, 提供了SVN、CVS的Ant调用接口, 实现了基本所有的版本库操作的Ant调用方法, 能方便地进行自动提交。

最佳实践

建立一站式的自动化流程, 建议使用单一的Ant构建文件, 确保每个子任务的清晰、独立, 子任务可以通过添加描述来显示操作进度和信息; Ant配置文件可以在头部定义变量和路径, 以便于修改维护; 可以使用分支、循环等技巧实现更丰富的自动化流程控制。

此外, 要注意中文编码, 合并等操作需要通过指定文件编码: encoding="UTF-8"; 有时需要对JS文件中的字符进行ASCII化, 可以使用内置的native2ascii标签进行操作。

结语

以上是使用Ant进行前端流程自动化的简单思路, 高度集成的自动化操作会产生一些不灵活的情况。

例如区分页面、应用的文件打包调用, 目前淘宝的许多页面已经使用CDN Combo, 用动态合并代替静态打包文件, 具体的应用还要视场景灵活选择。P

平板电脑的春天来了？

新平台

这个春天注定属于平板电脑，Apple的iPad2、Motorola的Xoom、HP的TouchPad、RIM的PlayBook，人称平板电脑的四大天王。并且四款平板的处理器和操作系统完全不一样，iPad2用的是自家的iOS 4.3和A5处理器，Xoom用的是Android 3.0和Nvidia Tegra2，TouchPad用的是webOS 3.0和高通的Snapdragon，PlayBook则是QNX和TI OMAP 4430。

平板电脑已成为操作系统厂商和芯片厂商的竞技场。ARM Cortex A9双核处理器成为平板电脑的标配，GPU则有PowerVR与Nvidia明争暗斗，除了Tegra2外的三家都使用了PowerVR的3D加速技术，Nvidia则靠自己在显卡领域的积累独步天下，形成了三英战吕布的格局。

强调自有操作系统，是高端平板区别中低端的显著特点，中低端平板是Android一统天下，而高端平板厂商采用了各自的操作系统。移动互联网的操作系统是作为开放式平台，承载第三方提供的应用与服务。这一点上，iOS是最成功的，Android紧随其后，webOS和QNX则没有搞清状况，还在靠自己为客户提供更多功能。

平板电脑的成败不取决于设备本身，而在是否拥有完整的生态系统。操作系统厂商只需要提供硬件平台、标准的开发环境和SDK、软件分发渠道和支付体系，剩下的应用、内容、服务都可以由第三方提供。为了保证利益最大化，操作系统厂商尽力扩大开发工具差异性，跨平台的应用移植

越来越难，开发者为支持多个平台而焦头烂额时，会选择只支持几个热门的平台。取悦开发者，将是各大平台厂商下一步的工作重点。

如何取悦开发者呢？首先，让开发者挣到钱，哪怕此目标遥不可及，但开发者始终相信，自己就是下一个幸运儿。其次，统一的开发环境和硬件平台，开发者不再为设备间的差异、不同的屏幕分辨率、不同的外部设备而疲于奔命，这一点上，iOS做得最好，Android最差。从Android 3.0开始，Google已经开始了战略收缩和规则重建，但覆水难收，Google的新策略能否得到OEM和芯片厂商的支持还是个未知数。毕竟当初很多OEM选择Android的主要原因就是想怎么改就怎么改。最后，为开发者提供各种各样的在线服务，移动互联网时代的开发者主要以个人和小团队为主，创意是他们的强项，而维护大量昂贵的在线服务器会成为他们不能承受之重。于是，为他们提供各种在线服务将是未来的热点，比如以Facebook为代表的身份验证和SNS服务，以OpenFeint和ScoreLoop为代表的Game Center服务，还有LBS服务、在线支付，等等。大公司会在移动互联网时代逐步退出用户桌面，转而通过云计算的方式为开发者提供在线服务。

国内还有无数山寨厂商试图在平板市场中分得一杯羹，但如果不理解平板背后的生态系统，iPad的神迹对他们来说，恐怕也只是水中月、镜中花。P



主持人：

马宁，微软最有价值专家，Windows Mobile开发者。

本月热点

新品发布

● Kinect销售超千万成为史上最畅销游戏外设

用户上一次为微软产品排队，恐怕还要追溯到Windows XP时代。Kinect在不到半年的时间里，创造了超千万的销售量，不可谓不是一个奇迹。不过Kinect的影响远远不仅限于体感游戏领域，已经有数个开源项目，尝试将Kinect的动作捕捉技术用于更广泛的领域。而微软也准备在稍晚发布基于Windows的Kinect SDK。

● Angry Birds Rio将在Amazon Appstore上独家发布

Angry Birds已经成为了一个响亮的娱乐品牌，与电影《RIO》合作的特别版Angry Birds Rio即将登陆Android。不过，这次并不是登录Google Marketplace，而是亚马逊的Amazon Appstore。随着Amazon加入战团，Android上的第三方软件市场将进入一个新的时代。在Kindle上积累了足够多的运营经验之后，Amazon是最有实力的第三方软件商店，虽然没有操作系统的支持，但是强大的运营、在线支付体系和云计算服务，形成了Amazon独有的优势。如果说第三方软件商店能够成功，那么非Amazon莫属。

2011程序员薪资调查报告

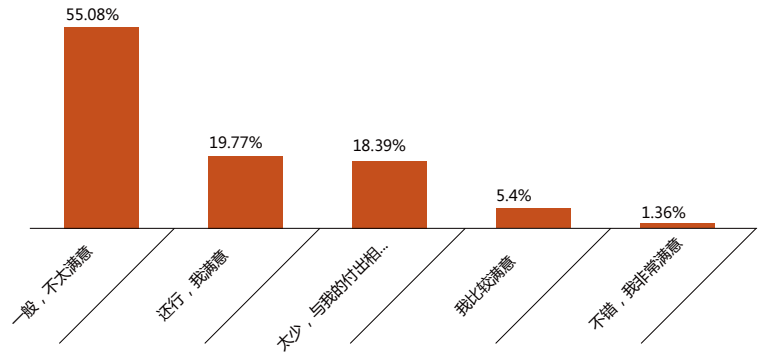
■ 记者 / 常政

自 从2011年初，CSDN在网上发起“2011年软件行业技术人员薪资大调查”（网址：<http://vip.csdn.net/2011/xinzi/main.html>）以来，引起了广大开发者们的热烈反响，短短两月时间内，近万名开发者提交了调查数据。尽管这只是中国百万开发者大军的一小部分，但他们所在的行业几乎涵盖了整个中国软件的产业链，他们的职位几乎代表了一个软件团队体系的每一个层面，而且“一叶知秋”，所以透过这些调查数据和变化，我们或许可以一瞰中国软件开发者的普遍生存状态，甚至可帮助开发者，更准确地定位自己在产业内的坐标。

2011：程序员的日子不算差

相信每个开发者在回忆当初高校毕业，加入滚滚求职大军的情形时，都能记得那番对美好生活的憧憬和闯荡世界的豪情。而在经济社会，判断成功的可量化方式无疑是薪资了，尽管有点世俗，但暂时也找不到更好的标准。那么现在，中国的程序员们，总体收入水平如何呢？首先我们

绝大多数程序员对当前薪资并不满意

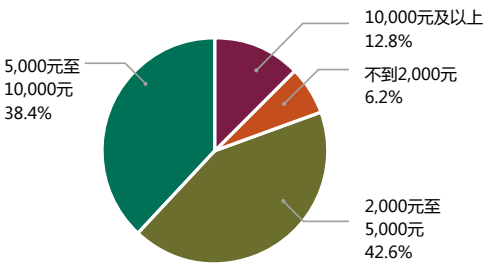


数据来源：CSDN网站2011年初程序员薪资大调查

看程序员们对薪资水平的满意度。

我们发现绝大多数程序员（近73%）对自己的薪资并不满意，这种普遍不满意的情绪有多少是主观预期过高，多少是客观生存环境造成的呢？我们需要做进一步的考察。

被访对象收入范围分布

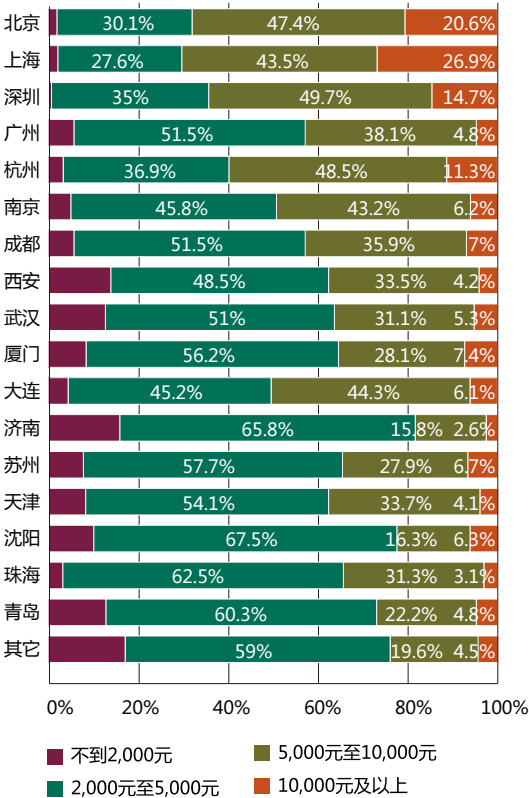


数据来源：CSDN网站2011年初程序员薪资大调查

我们按月薪大小把收入水平划分为四类：低收入（小于2000元）、中等收入（2000~5000元）、中高收入（5000~10000元）、高收入（大于10000元）。从调查数据看，来自中国17座重要IT城市的开发者们，占据绝大多数的是月薪2000~5000元，它在13个城市占据最高的比例，其中排前3名的是沈阳（67.5%）、济南（65.8%）、珠海（62.5%）。而北京、上海、深圳的开发者收入水平相对高些，这三座城市占据绝大多数的是月薪5000~10000元的群体。

如果仅依据国家统计局公布的数据显示，2010全年城镇居民家庭人均总收入21033元（月平均1753元），其中北京城镇居民人均可支配收入2.9万元（月平均2417元）。这两年国家经济相对比较稳定，估计2011年的情况也差不多。所以总的来说，2011年的中国程序员群体，在所处的城市里，和其他居民比，算相对

不同城市的程序员收入状况构成



数据来源：CSDN网站2011年初程序员薪资大调查
收入不错的了。

当然也有生存环境堪忧的，我们发现月薪少于2000元的群体，主要分布在济南（15.8%）、西安（13.8%）、青岛（12.7%）、武汉（12.6%）。而如果以月薪10000元（及以上）算高薪，排名前列的则是上海（26.9%）、北京（20.6%）、深圳（14.7%）、杭州（11.3%），杭州程序员群体的崛起令人关注，说明杭州近年来的信息化建设成就卓著。再回过头来看程序员的薪资满意度，我们通过交叉分析发现，程序员的满意度确实和薪资大小相关，收入越高，不满意的比例越小。但值得注意的是，不管哪个收入群体，都超过50%以上都表达了对当前薪资的不满，说明尽管日子过得不算差，但中国程序员们的幸福感普遍不高。

最佳跳槽次数，最好不超过3次

跳槽，一直是程序员们在职场生涯里所面临的热点话题。它是一把双刃剑，一方面会带

给你更多的视野和经历；另一方面，会降低你的企业忠诚度和所在企业平台的积累。所以很多开发者往往会面临是否跳槽的煎熬和苦恼。那么本次调查的数据显示，薪资和跳槽此数存在潜在的规律吗？

表 1 薪资/跳槽次数交叉分析表（百分比）

		您目前每月的工资属于以下哪一个范围？				合计
		不到2,000元	2000元至5000元	5000元至10000元	10000元及以上	
您换过几次工作单位？	从未换过	59.6%	46.6%	27.4%	18.0%	36.3%
	1次	18.8%	23.8%	25.9%	18.0%	23.5%
	2次	10.5%	15.2%	19.2%	16.8%	16.7%
	3次	6.8%	10.4%	16.8%	24.6%	14.4%
	4次	1.8%	2.5%	6.4%	10.3%	4.9%
	5次	1.0%	1.1%	2.7%	7.4%	2.5%
	6次	0.3%	0.2%	0.8%	1.9%	0.6%
	7次	0.0%	0.0%	0.1%	0.6%	0.1%
	8次	0.0%	0.0%	0.0%	0.1%	0.0%
	9次	0.1%	0.4%	0.0%	0.0%	0.0%
合计		100.0%	100.0%	100.0%	100.0%	100.0%

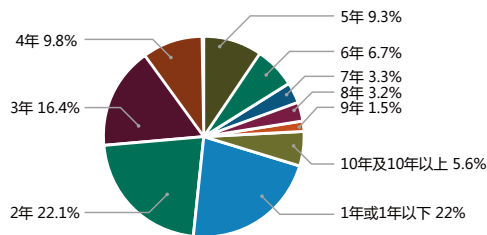
数据来源：CSDN网站2011年初程序员薪资大调查

从上面的“薪资/跳槽次数交叉分析表”，我们发现在四个收入群体中，“少于2000元”和“2000~5000元”群体中的绝大多数人都未换过工作，而从收入高于5000元的群体开始，有跳槽经历的人数显著加大。从工资高于10000元的高收入群体看，我们发现3次跳槽经历的人占据最多的比重，达到24.6%，但从第4次开始又急剧下跌到10.3%。所以从这样的数据结果可以看出，凡是有一定收入水准的开发者，基本上是有跳槽经历的，但跳槽的次数越多，并不绝对保证薪资高。数据显示跳槽次数存在一个“天花板”——3次。看来适度的跳槽有利于经验和技能的提升，但如果跳槽过于频繁，则不利专业的积累，自然在薪资上提升的空间也不大了。

技术菜鸟到牛人的距离，5年是分水岭

再来看工龄和薪资的具体量化关系，我们发现工作1~2年的开发者，工资在2000~5000元之间占据绝大多数，而工龄超过2年的，大多数人的收入达到5000元以上。同时我们发现薪资在5000~10000元群体在10年以内都基本处于一个稳定状态，没有明显增幅。而10000元以上的高收入群体，一个非常显著的变化是，前5年的人数增幅明显加快，但之后几年一直均处于稳定状态。

被访对象工龄分布



数据来源：CSDN网站2011年初程序员薪资大调查

表 2 薪资/工龄交叉分析表（百分比）

		您目前每月的工资属于以下哪一个范围？				合计
		不到2,000元	2000元至5000元	5000元至10000元	10000元及以上	
您从事软件开发工作多少年了？	1年或一年以下	19%	66%	13%	2%	100%
	2年	7%	61%	30%	3%	100%
	3年	2%	40%	51%	7%	100%
	4年	0.6%	29%	58%	12.4%	100%
	5年	1%	24%	56%	19%	100%
	6年	0.7%	14.8%	54.5%	30%	100%
	7年	0.5%	19.5%	46.6%	34%	100%
	8年	0.5%	14%	46.5%	39%	100%
	9年	0%	13%	50%	37%	100%
	10年及以上	2%	13%	38%	47%	100%

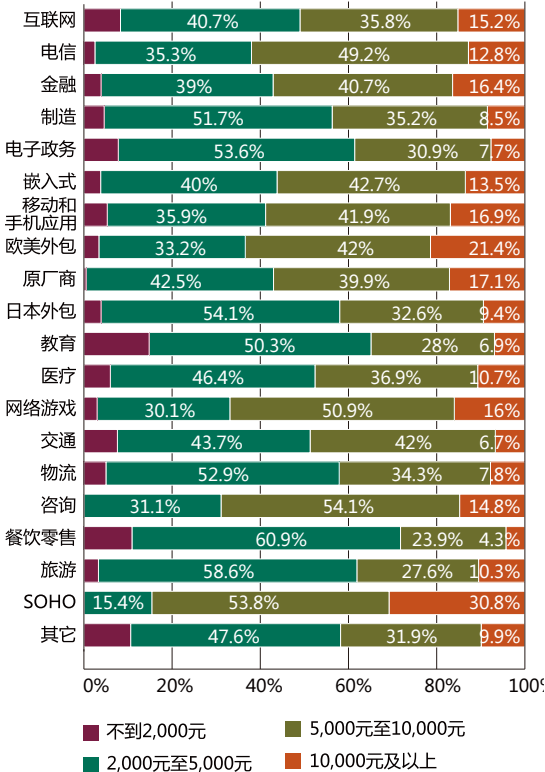
数据来源：CSDN网站2011年初程序员薪资大调查

所以，“3年（月薪5000元）”、“5年（月薪10000元）”是两个关键的分水岭。凡是月薪5000元以内的，随着工作年限的增加，人数递减；但随着年数达到3年后，月薪5000元以上的群体，人数开始显著递增。这不难理解，因为工龄的增加，开发者的工作熟练程度也越高，所以自然薪水也就高了。而工龄超过“5年”达到月薪10000元以上后的高收入群体，随后也基本开始保持稳定了。这说明，岁月对于技术开发者的薪资，同样存在一个瓶颈，并不是无限制正比例上升的。由此我们似乎可以推断，在中国软件行业，一个程序员菜鸟发展到业界认可的“熟练工”大概是“3年”，而“技术牛人”所需要的成长时间，大概是“5年”。

什么工作最赚钱？——不上班

“男怕入错行，女怕嫁错郎”。随着信息化在全社会范围内的渗透，所从事的细分行业的信息化发展水平和市场前景，已经成为决定

不同行业程序员收入状况构成



数据来源：CSDN网站2011年初程序员薪资大调查

开发者收入水平重要因素。那么作为开发者，选择什么样的工作，选择哪个行业的软件公司最有发展前景呢？调查结果令人诧异——自由职业者（SOHO）收入水准最高，超过30%的SOHO月收入超过10000元，月收入5000元以上的比例更是超过84%。但细想也在情理之中，有勇气做自由职业的开发者的，往往具备超高的技术水准和丰富的行业积累。

再看具体的细分行业，高收入开发者比例最高的领域是欧美外包（21.4%），看来中国软件本质上离“中国创造”的目标，还有很长一段距离。其次是原厂商（17.1%），这里的原厂商指的是诸如微软、甲骨文、IBM等软件巨头，其员工收入高并不意外。排名第三的是移动和手机应用（16.9%），这现象令人欣慰，毕竟未来就是移动互联网的时代。

从调查数据看，最不合适介入的是教育行业，小于2000元的低收入者比例接近15%，月收入小于5000元的接近65%。教育产业在国家属于公共资源，被严格管理，介入门槛比较高，再加上以“高考”为指挥棒的单一教学

导向，不容易衍生丰富多彩的信息化应用。此外，餐厅零售行业也是开发者需要谨慎选择的，低收入者10.87%，小于5000元的接近71%。不过餐厅零售业不像教育那样受到政府的严格管理，所以从乐观的角度，说明这个行业的信息需求没有充分挖掘。

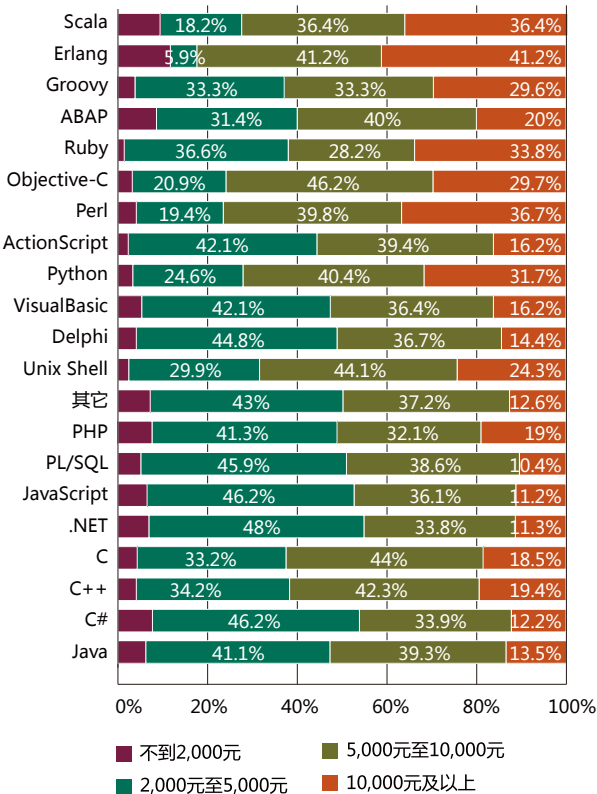
开发语言，选谁都一样

工欲善其事，必先利其器。开发语言、平台对于开发者来说，如同披荆斩棘的利剑。尽管对于顶尖高手来说，达到了编程思想、方法论层面的炉火纯青，可鸟瞰一切平台和工具，但对大多数初涉软件行业的程序员来说，熟悉哪种语言、开发工具往往直接决定了当下的收入水平和生活水准。从调查数据看，绝大多数开发者都使用JAVA，达到45.3%之高，其次是C#、C++、C、.NET、JavaScript，它们相对比较均衡，基本在25%左右（注：很多开发者往往实际会使用一种以上语言）。我们发现，C#、.NET开发者中，小于5000元的比例

最高，基本在55%。但不能因此说C#、.NET没有前途，因为另一数据发现，所有的语言，在5000~10000元的群体里，比例竟然惊人趋近，基本都在30%~40%之间，这说明不管选择哪个平台，只要达到“熟练工”水准，收入不会差太远。至于一些语言的低收入群体比例偏高，这和它容易学习，适合编程菜鸟上手有关，因为我们同时通过交叉分析，注意到工龄2年内的C#、.NET程序员小于月薪5000元收入水平的比例竟然高达80%左右，而工龄超过3年后，这个比例开始明显下降了。

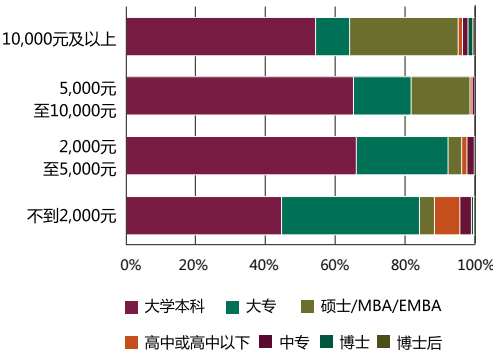
再看高收入群体，我们发现在使用Erlang、Perl、Scala技术的人中，高收入人群的比例较高，分别为41.2%、36.7%、36.4%。但我不建议大家一窝蜂地去学习这些语言，因为同时发现它们的样本量极低，分别是17、98、11，远小于近万份的总样本量，看来主要是物以稀为贵，会的人少，自然收入就上去了。

使用不同技术人员的薪资分布



数据来源：CSDN网站2011年初程序员薪资大调查

不同收入人群的学历分布



数据来源：CSDN网站2011年初程序员薪资大调查

结束语

在本次调查中，我们还结合英语技能、学历高低对开发者的薪资水平进行了分析，发现结果和常识是一致的，即英语、学历越高，获得高薪资的比重越大。总的来说，开发者要想过更好的物质生活，重要的是在专业性上下工夫，因为尽管随着城市（考虑消费水平在内）、技术平台、行业等外在条件的不同，低收入、中等收入群体的比例千差万别，但高收入群体的比例是基本接近的。

如果您还希望查看更多CSDN关于开发者的薪资调查数据，欢迎访问<http://hunter.csdn.net>，您可以自己进行地域、行业、工作年限等因素的组合，查看薪资分布状况。

2011推荐系统论坛随记

记者 / 付江

2011年3月6日，由淘宝网与ResysChina联合举办的“2011推荐系统论坛”在北京地质大学成功举行，《程序员》杂志记者受邀跟踪并全程报道了本次活动。本次大会无论从参会者的人数与覆盖面，还是从演讲者的整体水平，都大大超过了第一届（据说第一次活动只有30个人，绝大多数都是学生和草根）。而本次大会据主委会方面的介绍，作为免费的技术交流活动，报名的有1000多人，但为控制会议规模与质量做了一些人数控制，所以实际到会有200多人。Netflix Prize 冠军队成员Yehuda Koren、淘宝网数据产品部技术经理贾超（玄澄）、百度科学家张栋、淘宝网技术研发部广告算法组/一组经理项碧波（游龙）、前Google研究员王益、Hulu推荐团队技术负责人郑华为与会者带来了精彩演讲。

据悉，参加本次大会的代表中，有世界顶级学者、搞工程的、搞产品设计的、搞系统思维的……真是“啥都有”。

超过一半数据无法用模型解释

最先登场的Koren是本次大会的特邀嘉宾，作为Netflix Prize冠军队成员，远道而来的他没有令大家失望，不仅带来了一场干货十足的演讲，还带来了浓郁的以色列家乡风味的口音。Koren的演讲内容基本涵盖了这些年来他所有的相关工作，

不论是参加Netflix Prize时的基本模型与经验，还是时间效应如何融入到Model的探索；又或是如何构建快速的反馈系统以解决冷启动问题；还有像在Y! Answer那样的问答应用中如何融合多种数据属性而给出

Multi-channel推荐的问题，以及评论模型和推荐置信度的话题等等。总之，Koren的每个话题都分量十足，毫无保留，虽然侧重学术研究，但可以为商业应用所参考和启发的地方也很多。

在Koren对Netflix样本方差来源的分析中，超过一半是无法用模型解释，在可解释的成分中，“公认怎么样+个体打分松紧程度”占了一大部分。此外，他认为more user input > better algorithm，好的充分的数据比好的算法更重要，而用户long-term效应实际上很难被捕获。

淘宝网的嘉宾在上下午分别有一场演讲。玄澄的演讲比较概括，重点阐述了淘宝数据的力量。他表示淘宝正在推自动化数据运行的模式，这将逐步替代人工管理商品目录。此外，数据挖掘方面将采用更多数据可视化和交互的手段。其实如果从一定程度上来看，淘宝网上的交易数据实质上已经对中国城市居民的购买行为有了相当高比例的采样，从中分析得到的结果具有很纯粹的现实价值。玄澄列举了几个有趣的分析结果，例如：比较淘宝门票销售额可推论在热度上张学友>王菲>周杰伦；从喜糖购买人群的特征可推论有40%的男性是30岁以后结婚。如果这些数据能在一定程度上公开，必定是一笔巨大的财富。

拥有了如此庞大的用户行为数据，淘宝当然也会做出一些推荐与广告方面的尝试。下午游龙的演讲就主要分享了淘宝在电子商务推荐与广告投放方面的经验。游龙谈到了用户在网购不同阶段有不同的推荐需求，按时间顺序是发散、收敛、再发散的过程。这对于类似的电子商务推荐系统都有借鉴意义，按照用户的浏览习惯，具体可分为：四处闲逛阶段——可采用各种大跨度的试探性推荐，揣测闲逛者的消费意图；精益求精阶段——发现用户开始货比



Netflix Prize冠军队成员Koren做了一场干货十足的演讲

三家，可以把推荐范围逐渐缩小，加速消费者下订单、支付的进程；意犹未尽阶段——这时候可以施展“啤酒+尿布”的数据挖掘手段，推荐一些哪怕看似不相关，却在统计数字上有关联的商品，这往往会带来意想不到的效果。

此外，游龙还分析了各个网站之间的跨域访问需求，以及推荐系统反馈实时性的重要性。举个例子来看的话，据说在北京燕莎、双安、大悦城，好的导购能通过消费者嘴角一个0.13秒的翕动，在0.013秒内判断其消费意图并推荐合适的商品，电子商务的推荐算法必须要像这样的导购一样。如果慢了，消费者就把Tab关了，或者已经买完走了。

算法只占推荐权重的10%？

百度张栋的演讲比较偏技术，还带入了很多自己深层次的归纳和思考。让记者印象最深的便是他所描述的成功resys推荐系统的影响权重：UI/UE:40%，data:30%，knowledge:20，algorithm:10%。这也是当天活动中最具有争议的一个话题，引起了与会者的纷纷讨论。对于算法人员，需为外界的误读而捍卫自身的尊严，产品/设计者则以发现一个可以宣告自己工作价值的话题而喜悦。以至于到后来张栋还专门在微博中解释：“有人问我，这是不是让很多工程师沮丧。其实不是，因为这10%的事情，只有1%的人能做的好……另外，互联网上的任何一个应用1%的差距都非常的significant:Google 只比Bing好1%。”而至于这些数字是怎么得出来的，张栋在微博里引用了一个同学的解：“ $A>B>C>D$ & $A+B+C+D=100\%$ \Rightarrow A: 40% B:30% C:20% D: 10%，这是其中一个解。”

此外，张栋认为推荐系统是应用领域相关的，不可能跟搜索引擎一样有一个统一的框架和做法，即便Amazon内部不同大类的商品也需要有不同的推荐引擎。相对于对各种优秀算法的研究，张栋比较认同混合算法的威力，他谈到了自己在参加Netflix Prize大赛时候的感受，尤其是在Netflix Prize后期英雄榜上的队伍，用的基本都是混合的模型，包括成千上万个模型的组合。张栋坚持认为：一个好的算法无法打

败无数个Combine起来

的算法。在互

动环节，张

栋表示推荐

系统研究

未来的

三个发

展方

向分

别是：领

域专家系统、

迁移学习和

Temporal-bias。而他同样认为推荐系统将成为互联网应用的基础设施，这让现场的同学兴奋不已。

来自腾讯的前Google研究员王益贡献了一场关注底层技术主题演讲——

《Recommendation Using Large Scale Click Through Data》，特别是一个轻量级的C++实现类似于Mapreduce框架的内容，给与会者留下了深刻印象。

最后一个压轴的是来自Hulu的郑华，他分享了Hulu在视频推荐与广告投放方面的经验。据郑华介绍，Hulu算法团队有两大部分工作：视频推荐与广告投放。郑华认为商业上的因素会对推荐策略产生非常重要的影响，但是Hulu的广告投放则比较人性化，用户可以自己选择看哪种类型的广告；此外，在user feedback中，Hulu可用的显式的打分信息不多，只能更多地使用隐式的浏览信息。

写在后面的话

笔者全程参加了本次活动，现场感受最深的就是推荐系统正受到越来越多从业者的关注，或许真像Koren和张栋认为的一样，推荐系统将成为未来互联网应用的基础设施。推荐系统也发展成为一门综合学科，算法、用户行为数据、商业模式包括界面设计等等都在一定程度上影响到推荐系统的最后效果。而本次大会上嘉宾和参会者热烈讨论到的话题，也将在很长时间内引发从业者的思索。P

责任编辑：常政 (changzheng@csdn.net)



张栋描述的推荐系统影响权重，引起热烈讨论

NoSQL利器 MongoDB驾临中国

——MongoDB开发者会议引爆京城

记者 / 董世晓

在云计算热潮中，NoSQL（非关系型数据库）可能是最受关注的技术焦点之一。而MongoDB以其开源、功能丰富、查询语言强大、与关系型数据库相似等众多特点，成为最先得到广泛应用的NoSQL方案。

为了使国内技术人员深入了解MongoDB的技术内幕及其应用，CSDN和MongoDB开发商10gen于3月3日在北京联合举办了MongoDB开发者会议。本次会议，绝对可以称得上是一场MongoDB技术盛宴：10gen产品与工程高级副总裁Roger Bodamer和10gen企业工程与服务高级总监Alvin Richards一起详细讲解了MongoDB的核心技术，视觉中国CTO潘凡、Yottaa系统工程负责人吴向君、中国MongoDB社区创始人程显峰、alivenotdead联合创始人兼CTO王传仁分别分享了自己的实战经验。

参会人员有备而来

在活动现场，记者见到了数位来自盛大、淘宝、新浪等公司的技术专家，其中多人是从外地专程赶来参加此次会议。据统计，本次会议网上注册人数为600余人，最终参会人员总计将近400人。作为一场完全免费的会议，而且又是在工作日时间举办，能取得这样的成绩已相当不易。

提到参加此次会议的初衷，淘宝高级数据库专家宁海元（江枫）说：“从去年开始，我们就对MongoDB产生了浓厚的兴趣，并且在一些压力不是很大的环境中测试和使用这个产品。但在使用过程中，我们发现了一些问题。比如，1.6版本在刷磁盘时，数据库性能会急剧

下降，会有一个持续时间很短的尖刺。现在该问题还没有得到解决，至于是什么原因，我们想和MongoDB的专家现场交流一下。”

演讲内容引人入胜

Roger Bodamer从大局出发，结合实例介绍了MongoDB及数据库设计。内容涉及查询操作符、扩展架构、单表继承、一到多/多到多关系、树、祖先数组、变量键等。另外，他还提到建模的目的：插入、更新或删除时避免异常；扩展架构时减少重新设计工作；使模型对于客户来讲更具信息量；避免偏向特定风格的查询。

Alvin Richards则从复制（Replication）和分片（Sharding）两种功能对MongoDB进行了解读。对于复制功能，MongoDB遵循的是：从主库中读总是一致的、从从库中读最终是一致的、主库故障时自动容错移转、当节点加入集时自动恢复。而对分片功能，可以做到：不停机水平切分数据、写数据时自动平衡、命令被路由转至正确分节点（插入操作和更新操作必须有水平切分键；查询操作当有水平切分键时被路由至节点，反之分散集中）。

视觉中国是国内较早使用MongoDB的公司，其使用的情況必然是大家关注的焦点。潘凡现身说法，以视觉中国数据库从MySQL迁移到MongoDB为例，分享了他的成果和感受。经过迁移后，视觉中国的生产环境是：前端使用PHP/5.2x，后端Perl5.2.10+/Plack/Gearman，MongoDB用于回话存储、业务数据和用户上传文件、缩略图等多媒体文件。他将使用



会场座无虚席，彰显了业内对MongoDB的兴趣

MongoDB的最大收获归结为：灵活和自由。MongoDB可以轻松面对不同数据源，随时调整变化的需求。对于正在开发创新型互联网应用的公司来说，MongoDB的这些特性，正适合它们快速开发产品、快速迭代，适应用户迅速变化和更新的种种需求。

吴向君谈到，与其他公司一样，Yottaa对MongoDB也是经历了“从观望到运营”的过程。在这个过程中，Yottaa的收获是：必须提前考虑分片；升级时复查所有慢查询，并添加合适的索引；在生产环境中，必须小心添加索引；避免慢写操作等。

程显峰以他自己的一个项目——MongoKoans，详细解释了怎样用Ruby巧学MongoDB。谈到alive.cn使用MongoDB的原因，王传仁说：“MongoDB对于不同的数据源可以使用灵活的数据模式；在处理大量数据时，MongoDB有良好的伸缩性；MongoDB能够使用重复的数据集来保持数据在墙内外的一致性。”谈到未来，他认为开放API、自动主题提取和多语言的实时趋势分析都将是大势所趋。

优化之路尚需继续

MongoDB作为一种文档型数据库，非常适合快速开发。在开发初期并不需要考虑太多文档的结构，而且与MySQL相比，MongoDB性能高出很多。

作为一个比较新的数据库，MongoDB必然还会存在一些问题需要解决和优化。盛大创新院郭理靖就表示：“我感觉MongoDB在两方面做得比较好。一个是高并发问题，可以有效缓解请求压力过大的问题，另一个是可以自动热

备数据，轻松解决容灾问题。但是，MongoDB仍有些问题需要改进，一个是备份和Rollback（回滚）做得还不太理想，Sharding还有待考验。另外，在内存使用吃紧后，还存在性能急剧下降的问题。”

在Twitter宣布放弃Cassandra后，曾使得业界一度对NoSQL的前途产生了些许担忧。但就目前情况来看，NoSQL还是深得人心。而MongoDB作为NoSQL的代表之一，其前景如何，让我们拭目以待。P

关于MongoDB

MongoDB是一个高性能、开源、无模式的文档型数据库，它在许多场景下可用于替代传统的关系型数据库或键/值存储方式。Mongo使用C++开发，提供了以下功能。

面向集合的存储：适合存储对象及JSON形式的数据。

动态查询：查询指令使用JSON形式的标记，可轻易查询文档中内嵌的对象及数组。

完整的索引支持：包括文档内嵌对象及数组，同时提供查询优化的功能。

查询监视：Mongo包含一个监视工具用于分析数据库操作的性能。

复制及自动故障转移：其数据库支持多形式的数据复制，提供冗余及自动故障转移。

高效的传统存储方式：支持二进制数据及大型对象（如照片或图片）。

自动分片以支持云级别的伸缩性：自动分片功能支持水平的数据库集群，可动态添加额外的机器。

责任编辑：常政（changzheng@csdn.net）

微软 MVP 萧涵忆： WP7 将和 Android、iPhone 三足鼎立

记者 / 杨东杰

最近几年，微软一直在推动两岸三地的 MVP 互通交流。本次 MVP 系列专访的主角萧涵忆就来自台湾。作为一名资深的 MVP，他已连续 7 年获得 MVP 荣誉。由于在 MVP 相关工作中的出色表现，微软称许萧涵忆为“微软产品在用户和合作伙伴信息交流环节中的关键人物”以及真正的“Windows Phone 专家”。在本期访谈中，他将讲述自己的 MVP 体验和对 Windows Phone 未来发展的看法。

记者：您在 2008 年前担任了 5 年 Office System 方向的 MVP，之后为何转到 Windows Phone 方向？

萧涵忆：很多人问过我这个问题，我觉得或许是“路见不平，当‘MVP’相助吧”。毋庸讳言，iPhone 与 Android 的确为用户展现出一致性的亲和力。但是微软不论是在 Mobile 系统上的超强商业性能或是 Windows Phone 7（以下统称 WP7）上的优异整合界面上，许多原创设计与巧思至今都引领着业界潮流。我很奇怪这么优秀的产品竟然没有出现与之成正比的用户群体，所以便转换跑道至 Windows Phone 领域，希望能为 Windows Phone 尽些心力。

记者：您认为诺基亚和微软联盟后，WP7 的发展前景如何？是否会与 Android 和 iPhone 成三足鼎立之势？

萧涵忆：事实上，我对此非常乐观！只要使用过 WP7，大多使用者的共同体验都是爱不释手。透过各种整合中心（Hub），使用者可以随心所欲地尽

情享受指尖操控带来的愉悦感。

当然现阶段 WP7 还有一定的改进空间，但微软长期以来也一直在精益求精。目前微软 Marketplace 的应用数量正在急起直追，不过微软固有的 Mobile 开发群体长期以来主要都投入在商业与生产力工具的研发，相对在情境、娱乐、游戏等方面的经验稍微不足。最近微软与诺基亚的结盟，正面效益指日可待，有了 Ovi Store 的加入，不仅两个阵营的开发者可以互相切磋、激荡火花，更直接的效益是直接强化了 Marketplace 的内容，譬如地图服务的本地化语言问题。

其实微软的产品在操作设计与前瞻思维上都是表现很优异的，Windows Mobile 在 6.5 版本时，许多技术都是当时 iPhone 与 Android 所追求的，譬如网络公用以及 IE 的安全性（兼容大多数企业网站）等。到 WP7 则在设计界面上摆脱了目前其他阵营的设计思维，以 Hub 为单元，使用者可以在美观的底图上尽情滑动，体验资讯整合的乐趣。其他特性如让 Facebook 直接与手机结合，无须另外安装 App 等也是创新的设计；在 Office 上除了强调编辑能力大幅提升外，结合云端也展现了微软强烈的企图心；XBOX 的游戏整合更是让许多人跃跃欲试。基本上我觉得 WP7 是一款操控流畅，在个人资讯、工作与娱乐都有优异表现的手机平台。

记者：从开发者角度，您认为 WP7 有哪些其他平台不可比拟的特性？对于有兴趣投入 WP7 平台的开发



萧涵忆认为 WP7 的优势还不为用户所了解者，您有何建议？

萧涵忆：我本身不算是开发者，所以回答这个问题有点班门弄斧。不过微软长期以来的优势是在生产力工具、商业流程、资讯安全、个人资讯管理（Outlook）等方面。这个方向的应用在 Windows Marketplace 上还有很大的成长空间，由于这些工具有助于整合知识工作者的工作与生活，市场需求很大。

另外 WP7 目前在云端服务与体感操控技术的优势也能有效吸引其他阵营的用户转身拥抱 WP7，所以，结合这两个特性的应用也具有开发效益与后续发展的机会。

不过不论是哪种应用，值得所有开发者注意的是，在基本的功能能够流畅运行后，产品的美工设计也要非常重视。P

学生兴趣是信息技术教育的切入点

——北大附中信息技术特级教师李冬梅专访

记者 / 陈秋歌

国内IT技术正以飞快的速度向前发展，在欣欣向荣的信息技术领域，有一批默默无闻为该行业发展贡献着自己青春的人，他们虽未站在信息技术发展的最前沿，却为该行业输送着源源不断的技术人才——这就是信息技术教育工作者。北大附中信息技术特级教师李冬梅就是其中的一名。

攒机组网，搭建教学环境

1982年李冬梅从北大数学系信息专业毕业后，便开始在北大附中担任中学信息技术教师。当时北大附中的一台计算机也没有。李冬梅跑遍北大所有计算机实验室，只找到了两台教授们从国外带来的微机。为了让学生有足够的计算机可以操作，她便利用周末带领学生组装机，首批组装了386。回忆起第一台386的组装过程，李冬梅说：“当时带领十几个学生把买来的所有零件按示意图一点点尝试组装在一起后，一接电便冒了烟，当时学生都被吓坏了。后来经过检查才发现主板的电源线插反了，重新组装尝试后，我和学生联合组装的第一台386机器也就诞生了。”国家有了网络之后，李冬梅便着手带领学生搭建校园网，北大附中的第一个总线型局域网便是李冬梅着手组建的。

回想起三十年的教学历程，李冬梅认为作为一个中学的信息技术老师，要付出比其他科目教师更多的心血。不仅要教程序设计、编写教学软件，还要设计校园网、规划校园信息化管理的蓝图，甚至要会装机器、修机器。

敬业乐业，探索教学模式

“学生眼中的信息课是如此的真实有趣，是因为背后融合了好大的一个系统：既有完整的知识系统，也有完善的作业系统，还有具体的评价系统。”这是同行教师对李冬梅教学成果的评价。所谓的背后系统指的是李冬梅根据多年教学实践总结出的一套符合中学信息技术学科的课堂教学模式，该模式具有三个核心要素。

- **教学环境的搭建与建设。**李冬梅主张将教学由教室迁移到机房，并首次采用Windows域管理模式管理机房，方便了老师与学生的沟通。

- **教学过程的设计。**她认为抓住学生兴趣，是搞好信息技术教育的关键。在课堂上，李冬梅不会一上来就硬丢给学生一堆理论和案例，而是提前为所讲知识设计好故事情境，引导学生去思考，去学习。比如教学中为了说明伪随机函数的含义及“Randomize Timer语句”的作用，她会设计一个“猜数游戏程序”作为情境，引发学生思考。当学生对计算机如何产生随机数产生疑问时，李冬梅便趁机告诉他们有一个随机函数“RND”在背后“作祟”，而当他们对每次产生相同的随机数而不解时，她便自然而然地引出“Randomize Timer语句”在这个过程中所起的作用。

- **评价激励机制。**李冬梅自行开发了一套网上评价系统。通过该系统，学生不仅可以看到老师对其作品的正面、肯定的评价，同时也可以看到系统中“荣誉榜”上的优秀作品列表。“榜上



李冬梅认为学生兴趣是信息技术教育的切入点有名”便成了学生争相设计出优秀作品的动力。

努力付出，换来丰收成果

正因李冬梅独特的教学方式，激发了众多学生的学习热情。比如来自杭州千岛湖的管雷，在入学北大附中高中不久，便迷上了李冬梅带的信息技术课，并整天跟着李冬梅学程序设计，毕业后保送到北大数学系。2007年在方正工作期间，因数字化音视频控制技术及应用项目，获得国家科学技术进步奖二等奖。据李冬梅介绍，仅1994年她所带的实验班23名学生中就有3人出版了计算机个人著作，其中一人所写的Windows 95教材发行近八万册，四人考入北大清华的计算机系。

百年大计，教育为本。国家信息技术的发展离不开像李冬梅一样为信息技术付出青春的教育工作者们，他们虽没有该领域领军人物一样的光环，却是信息技术大厦坚实的地基。P

互联网的未来：现实与虚拟结合

——记CSDN CTO俱乐部3D互联网专业委员会首次线下活动

3月16日，CSDN以“虚拟世界真实财富——生活在3D互联网时代”为主题的第39期CTO俱乐部活动在北京中科院计算所大楼举行，这也是CTO俱乐部3D互联网专业委员会成立以来的第一场线下活动。本次活动围绕3D虚拟世界等相关话题展开，有近50名CTO俱乐部会员参加了本次活动。

3D将是未来互联网的主角



赵刚表示伴随硬件、带宽的飞速发展，3D将成为互联网的主角

HiPiHi公司技术副总裁赵刚担任本次活动的主持。他表示互联网经历了1.0、2.0时代，正在向3.0时代发展。在3.0时代，3D将是互联网的主角，并引领互联网从信息互联网向体验互联网方向发展。

互联网由1.0发展到2.0时代，信息交流模式完成了由单向传播到双向互动的转变。在未来互联网3.0时代，信息将获得全方位的互动，用户不但可以从中获取信息还可以获得身临其境的现实体验。

虽然互联网发展之初就已有支持3D的技术，但发展比较慢，“随着硬件、带宽的飞速发展，互联网支持3D的技术会越来越好，届时大家会觉得互联网中3D才是主角。”赵刚认为。



许晖认为3D虚拟世界将给人们带来一种全新的互联网体验

3D虚拟世界的发展面临众多困难

HiPiHi公司CEO许晖在主题演讲中，分别从对3D虚拟世界的看法、3D虚拟世界发展过程中面对的困难和挑战及对未来3D虚拟世界的设想三个方面分享了他多年来对3D虚拟世界的探索经验。

所谓3D虚拟世界，是指采用互联网技术、三维数字技术、游戏引擎技术、即时

通讯技术等手段创造出的与现实生活相似或部分相似的、用户可以进行虚拟活动的空间。虽然3D虚拟世界外表酷似游戏（此处游戏特指大型MMORPG游戏），都需要登录互联网，都需要下载客户端，都需要创建人物作为这个世界的替身……但它与游戏却有着核心区别：游戏是玩家在事先设计好的特定情节和环境下进行活动、体验，从升级、打怪中得到快感，用户只是游戏的体验者；而3D虚拟世界是让用户根据自己的喜好打造属于自己的情节、环境，并可邀请其他人共同进行创造和体验，用户是这个世界的创造者。

许晖认为，在3D虚拟世界中，人们的视觉、听觉及触觉等感官都可以被调动起来，从而获得一种全新的互联网体验，即所谓的人联网。2005年许晖创办了HiPiHi公司，将其定义为3D虚拟世界的平台。“我们试图把HiPiHi打造成一个世界平台，并构造出3D虚拟世界的自然法则。人在其中可以改变现实世界中所带来的约束，重新缔造一个真实的自我。”许晖说：“然而在尝试的过程中却遇到了诸多困难。虚拟世界要求对所有事物进行仿真，且所有事物都是不可预知的，需要开放给用户自我创造，这就对技术提出了极大的挑战，似乎是一个公司无法完成的。同时如何把3D虚拟世界跟用户真实的需求完整的结合起来，也是一个非常困难的现实问题。随着带宽、硬件配置的进步，相信此问题可以得到解决。”

为了翻越3D虚拟世界发展过程中遇到的重重大山，许晖表示正计划把HiPiHi平台平放出来，以推动者的身份，与业界联合起来共同探索、共同推动3D虚拟世界的发展。

Avatar是3D虚拟世界的核心概念

中山大学哲学系教授、博士生导师翟振明表示Avatar是虚拟世界的核心概念，在虚拟世界中每个参与者都有一个三维形象的Avatar（替身）。视觉中的3D对象可以把人的听觉、触觉及其他感觉整合在一起，参与者通过在线控制Avatar，便可在共同创建的网络化3D感性世界中进行交流互



翟振明表示虚拟世界应以Avatar为中心，将其作为抵挡一切外来侵入的核心

动，从而获得一种浸蕴其中的在场体验。

当前的虚拟世界，只是作为人与人之间相互交流的场所，可称作人联网。如果把当前的物联网与人联网相结合，将会发生什么呢？翟振明分析称：“人联网只能让人获得丰富的体验生活，并不能满足人们的生活所需。如果把物联网整合到人联网中，人通过Avatar来操作物联网监控下的所有事物，这样人联网就成为操作物联网的一个大界面。可以设想在这种情况下，人们只要在虚拟世界中通过遥距操作技术就可以搞定现实中的一切事务。此时人联网加物联网加遥距操作技术就构成了扩展现实（Augmented Reality），它不仅是交往与体验的世界，更是操控自然因果过程的实践空间，亦即人们为生存和发展而劳作的地方。这将对人类生活产生特别大的影响。”其实主从机器人就是该设想的一个现实版，现已用于医疗领域。

然而当前多数人认为人联网是玩物丧志的地方，物联网才能真正给国家带来GDP。随着人联网发展壮大后，在整合过程中，是人联网主宰物联网，还是物联网控制人联网将会成为一个很有争议的哲学问题。翟振明对此表示人联网和物联网整合后，每个Avatar都应该被赋予一个私人空间，就像在现实世界中每个人都有一个人不能被他人任意侵入的物理空间和信息空间一样，它必须是独立的，不能为某些大中心或物联网所操纵。所以虚拟世界在一开始制定标准时，就应该以Avatar为中心，为每个Avatar设定独立、隐蔽的数据包和个人防火墙，将其作为抵挡一切外来侵入的核心。



苏兴华表示“3D虚拟体验无所不在”是PLM2.0的核心理念

3D逼真体验是PLM2.0的核心特色

Axis3D 产品总监苏兴华在演讲中列举了当前的热门话题，比如：虚拟世界、增强现实、立体电影、3D Online、社交网络和微博、3D游戏、云计算、移动。同时分享了他总结的Web3.0新三国论，即微软与苹果代表的操作系统、谷歌地球、3DS达索析统组成了Web3.0

时代的新三国。

达索析统是3D技术及产品全生命周期管理方案（PLM）的提供者。PLM2.0是达索析统中一个很重要的概念，“3D虚拟体验无所不在”是其核心理念。PLM2.0提供的三维逼真体验可让客户通过Web2.0工具介入产品开发过程中，无论是设计工程师还是客户都可以创建产品。采用该技术，企业可充分利用世界各地的各类用户群体的集体智慧。

苏兴华表示所有的产品设计都应该通过现实模拟来预估其设计的合理性。利用PLM2.0，不仅可以模拟看得见的东西，也可以模拟看不见的东西，比如核辐射、地震波、瘟疫扩张速度等。相比于提供一堆难以理解的报表，3D可以带来更加逼真的体验。

在演讲中，苏兴华还演示了达索网站上的一个3D交互式应用程序。在这个应用程序中，用户可以按个人爱好设计自己的车辆，在设计过程中可以看到车里的众多细节，并获得一种身临其境的体验。厂商可以根据用户提交的设计车型对现有车型进行改造或设计新的车型。



本次活动以3D虚拟世界为核心主题，有近50名CTO俱乐部会员参加。

结语

2005年美国著名虚拟世界Second Life的成立及迅速盈利，引起了业界对3D虚拟世界的关注并开始刺激中国市场。同年HiPiHi的成立，以及随后出现的Novoking、uWorld、MWorld等都投入到3D虚拟世界的浪潮中，开始了实践与探索。虽然让虚拟世界真正走入现实生活，完成现实与虚拟的完全整合还有很长一段路要走，但这些“弄潮儿”相信未来的互联网领域必然会呈现出虚拟世界和Web页面形态的互联网长期共存的局面。（记者/陈秋歌）

分兵策略应对高速发展

驴妈妈CTO张勇认为应对公司业务的高速发展，应当采取分兵策略：一组人维护现有格局，另一组人开发新的平台。只有这样，才能保持高速发展，合理控制现有需求。



张勇
驴妈妈CTO

记者：您为什么选择加入驴妈妈？

张勇：有很多因素，我觉得主要有三个方面：

势：每一个企业的存在，首先是由于有市场空间。一个有潜力成为商业航母的企业，当然特别吸引人。驴妈妈给予了我与公司其他管理者，乃至资本市场足够的想象空间，所以我相信驴妈妈的发展速度必将非常迅猛。

人：驴妈妈早期的管理者都是在旅游行业浸淫多年的旅游业规划者和设计者。他们掌握的丰富资源，为驴妈妈的早期发展、与景区的深度合作奠定了商务基础，同时基于对旅游的深度理解，更能够从旅游者的角度设计产品，更好地关注用户体验。他们的经验，为我们能够推出一系列用户体验较好的产品奠定了坚实的基础。再者，驴妈妈的创业者都是一群有梦想，同时敢于执行的实干家。

时：驴妈妈选择采用电子商务的模式进入IT业，顺应了时代的潮流，现在的商业环境也已使得大众接受了电子商务这个模式。

记者：您加入驴妈妈之后如何规划整个公司的产品布局？

张勇：加入驴妈妈后，我分析了驴妈妈的整体系统平台，摆在我面前有两个难题：如何合理控制需求；如何重构既有系统。公司的业务在高速发展，既有的产品需要不断完善，同时新的产品也需要上线，这几乎是所有高速发展的企业必须面对的现实，我的方法是兵分多路：一部分人专注于既有系统，另外一部分人布局、开发我们未来要用的新系统。经过一年的改造，我们已经取得了一定的成效。

记者：驴妈妈CEO洪清华曾经说过：“驴妈妈的速度是看不见的”。作为CTO您是如何用技术来实现他这样的构想的？

张勇：驴妈妈所作的是一个基于Online to

Offline的商业模式，速度不仅仅体现在线上的系统，也体现在线下的合作。技术是提升公司竞争力的手段，为了支持公司的高速发展，在公司成立初期主要是打造一个支撑平台，将网站的各个功能模块分而治之，使得每个模块都能够独立发展和完善。在技术层面上，我们努力贯彻系统的垂直划分、可扩展性和可延续性。

记者：驴妈妈网第二版在风格、架构、功能方面都进行了很大的改进，并且融入了Web2.0功能。请您谈谈在这次转变中，都应用了哪些之前不曾用过的技术？

张勇：我们更多地采用了AJAX，同时较为彻底地变更了驴妈妈的早期基础框架，使得前端用nginx做静态页面处理和动态请求转发，同时后端采用Tomcat均衡提升动态请求的可靠性和稳定性。在开发框架上也由SSH框架变更为基于Struts+Spring+Ibatis的框架，目的是能够更好地控制对数据库的请求。并且，采用Lucene做搜索引擎，将自有词库引入，确保搜索的准确和有效，使得之前直接依赖数据库的搜索对系统产生的压力得以化解。在这次转变中，我们还完成了功能模块的垂直划分，使模块间依赖Web Service接口进行功能交互，从而确保各模块能够独立的升级、开发和部署。

记者：请您展望一下旅游交易平台的未来。

张勇：旅游交易是一个庞大的市场，但目前还处于群雄逐鹿的时代。未来在这个市场上一定会有巨人出现。未来的交易平台更多的会演变成：前端为游客提供预订服务的平台（如驴妈妈的B2C网站）；后端是景区自身ERP系统的外延和开放，提供分销的接口用于对接。这些直接为客户服务的平台，加快了信息流通，提升了合作效率，从而完成了信息全流通的闭环，为旅游行业电子商务化奠定坚实的技术基础。（记者/张雪峰）

HTML



来临!

■ 策划 / 本刊编辑部

作为下一代互联网标准，HTML5的地位举足轻重，已成为大家的新宠。2009年，W3C转向HTML5，如今，标准专家们把所有令人兴奋的、新奇的Web技术都整合进HTML5，而这一切都不需要插件。

Chrome、Safari、FireFox、Opera等浏览器也积极完善着HTML5实现，如今，微软凭借着IE9的正式发布也加入到标准的行列，这一切意味着：HTML5时代来临了。

无论你是重视互联网产品，还是关心用户体验，或者专注于前端开发，HTML5都将是不容错过的技术热点。

本期封面报道，聚焦HTML5技术，邀请从事一线开发的工程师，探讨HTML5技术应用，展现HTML5不可限量的应用前景，探讨HTML5你应该知道的一切，并结合实战，剖析HTML5与2D游戏开发、Web浏览器、视频开发等多方面的话题。

虽然目前HTML5还很不成熟，实践者们还面临着许多的难题，但作为能让互联网更美好的技术，HTML5仍然值得期待！

HTML5技术应用分享

——第八期TUP系列活动报道

■ 整理 / 佳琦

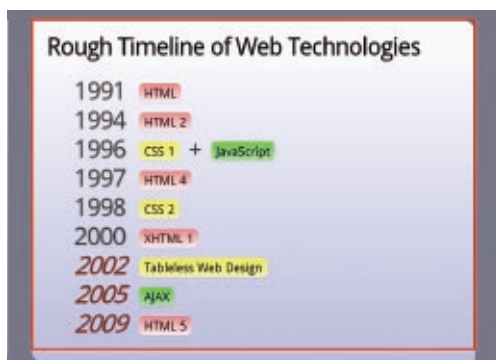
3月19日，第八期TUP系列活动在北京银科大厦举行，TUP是由CSDN、《程序员》杂志和JavaEye联合策划组织的线下活动，邀请业内知名专家分享IT产品背后的技术和用户体验故事。本期主题为“HTML5技术应用分享”。活动邀请了Google工程师高寒蕊、酷6开发部项目经理黄后锦，他们分别就HTML5的不同方面做了精彩介绍。



演讲人高寒蕊，是Google开发技术推广工程师。她全面介绍了HTML5的新技术。

HTML5技术介绍及应用

Web技术诞生到如今经过了20年时间（如下图），这相当一个人从出生到青年的时期，Web发展也进入全盛时期。



Web技术发展时间表

2009年是HTML5技术的里程碑。这一年，W3C研发重心又放在HTML5上（之前是XHTML2），同在这一年，微软宣布在IE将支持HTML5技术。而今天我们所谈论的HTML5技术，其实包含了HTML、CSS和JavaScript等内容。下面就HTML5包含的一些新技术介绍给大家。

存储与离线

存储包括包括三项技术：Web Storage、Web SQL Database、IndexedDB。

Web Storage包括两部分：localStorage和sessionStorage，两者最大区别是关掉标签页浏览前重新启动这个存储还在不在。

Web SQL Database使用语法与SQL基本一

样，但2010年W3C暂时停止对这方面技术的支持，因为SQL本身不是非常完善的标准，把Web标准建立在此基础上是不牢固的，在W3C没有找到好的方案之前，暂时停止。但各个浏览器并没有把支持接口移除，现在还是可以继续使用。

第三项存储技术叫 IndexedDB，这个名字非常形象地表达出这项技术核心是带索引的数据库。

实时与交互

实时和交互方面的内容，首先是Web Workers，可以在用户操作的同时，后台运行任务，任务执行完毕，则会通过消息的方式告诉主进程，主进程通过响应这个消息事件进行相应的处理。

WebSocket向客户端推送消息，主要解决长连接的问题，由它决定在合适时间将消息主动推送给用户。还有Notifications，可以在桌面弹出一个提示框，跟客户端程序窗口功能差不多，并且不要求页面是活跃状态的，当浏览器最小化，甚至被关闭的时候（这需要使用到Chrome的backgroundapp技术），Notifications一样可以弹出来。

文件与硬件交互

文件和硬件的交互方面，不仅仅可以在页面上对元素进行拖拽，也可以在桌面和页面之间进行拖放，一个比较好的例子是Gmail可以通

过拖拽上传/下载附件。

进行数据存储的时候，如果文件比较大（大于5MB），可以使用FileSystemAPI，它使用的是SandBox文件系统，不会对用户的操作系统产生破坏。

Geolocation已经非常火了，在LBS领域的应用很好地实现了真实生活与虚拟世界的融合。

而手机上的导航应用，Speech Input绝对比用手输入快很多，虽然现在准确率不是特别高，但是发展前景广阔。

语义化标签

以前讨论语义化标签所实现的功能不过

“DIV”实现功能，但读取网页的除了人，还有读屏软件、爬虫等等，怎么更好理解网页是非常关键的，语义化就能解决这个问题。

图像与多媒体

以前我们制作网页图像，经常直接嵌入图片，或者借助动画技术，现在可以使用原生的JavaScript的语言在Canvas画图，也可以直接在HTML页面里通过SVG做出图形。

其他

Webfonts可以做出非常漂亮的字体，甚至火星文。Text wrapping，会在根据文字自动省略显示不了文字，并加上省略号。

HTML5在视频网站的应用

说到视频，HTML5需要直面的是与Flash的竞争关系。在PC端，HTML5短期内无法撼动Flash的霸主地位，而在移动设备上，Flash的安全性和执行效率是最大的问题，这恰恰是HTML5的优势所在——手机上用HTML5看视频时电池续航能力是Flash的一倍！所以下面主要讲在移动客户端开发HTML5视频应用。包括属性、特性、方法、事件等几个方面。

首先，我们需要简单区分一下属性（Attribute）和特性（property），写入HTML标签里面的是属性，通过JavaScript可以访问的属性称为特性。

其中，poster属性，用来显示视频缩略图，在视频正在加载或数据无效时使用，暂停时是否显示则取决于各个浏览器的不同实现。

autoplay属性，在PC浏览器上可以实现，但是在手机上，为了节约资源不能自动播放，必须手动设定。在iOS 3.2版本可以通过创建一个事件响应，模拟单机实现自动播放功能，但是iOS 4.2以后这个功能取消了。

controls属性，在各个浏览器上的表现是不一样的，对此W3C给出了说明，由各个浏

览器自己决定。

error特性，大家都知道通过网络的应用多是靠不住的，所以很多应用统计时会用到这个特性。

currentsrc特性，通过此特性，可以访问到当前播放的视频地址。

接下来谈一谈HTML5 video方法。

其中，load方法，可以通过它重新获得视频地址，重新加载。

canPlayType方法，用来判断浏览器是否支持指定的视频格式。

HTML5 video事件比较多，经常用到有如下几个。

progress事件，通过它可以获得正在播放的视频进度。

play事件和playing事件则分别指“即将播放”和“正在播放”，需要明确区分。

volumechange事件，可用于PC端浏览器的音量控制，目前在移动客户端还不能使用。

timeupdate事件，播放进度发生变化的时候，时间会改变，据此可以制作进度条。P



演讲人黄后锦，是酷6开发部项目经理。他介绍了HTML5在视频网站的应用。

浏览器对HTML5的兼容性分析

■ 文 / 崔康

随着HTML5技术的普及和推广，Web开发社区逐渐开始尝试在RIA（富互联网应用）中使用HTML5，这种实践主要优势在于开发人员能够通过HTML5的各种技术以一种前所未有的统一、简洁、无插件依赖的方式实现炫目、复杂的Web应用，HTML5的优越性已经得到了开发社区的普遍认可。不过，就像其他Web技术诞生时的情况一样，作为新生儿的HTML5在各种Web浏览器中的实现程度或者说平台兼容性一直是Web开发人员的心病。本文将介绍Web浏览器对HTML5技术的兼容性，希望能够帮助Web开发人员在应用HTML5技术时充分考虑到平台差异性。



作者简介：

崔康，目前就职于IBM中国研发中心，从事企业级Web应用的开发测试工作，出版多本译著，感兴趣的领域包括Web性能优化、浏览器技术等。

从2006年WHATWG和W3C决定合作制定HTML5标准以来（预计2012年成为候选标准），浏览器厂商都不甘落后，一直在努力实现更多的HTML5技术，不过由于各个厂商的步伐不一，导致HTML5在浏览器中的实现程度参差不齐。在这里我们将综合两种HTML5兼容性测试工具包HTML5Test和Modernizr的测试结果来分析一下Web浏览器的支持程度。

考虑到HTML5标准的制定原则：新特性基于HTML、CSS、DOM以及JavaScript；减少对外部插件的需求（比如Flash）；独立于设备等，我们选取了HTML5的几项主要特性来评价浏览器系统。

- 视频播放标记video

- 音频播放标记audio
- 绘图文标记canvas
- 表单控件标记url、email等
- 本地客户端存储

浏览器（版本）选取

由于目前市面上的浏览器品牌和版本繁多，我们难以覆盖全部产品，因此在这里设定测试浏览器的选择原则为：主流品牌、主流版本和最新版本（正式发布）。依据StatCounter发布的2011年2~3月份浏览器市场占有率统计分布情况，主流品牌如图1所示。

从图1中可以看出，微软的Internet Explorer以其混搭的IE6、IE7、IE8、IE9等多个版本占据了榜首，开源浏览器Firefox紧随

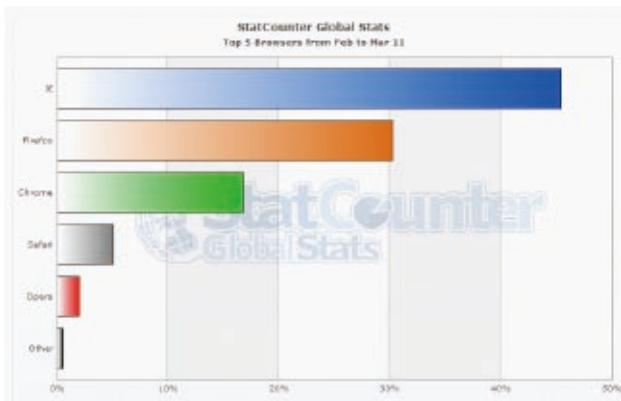


图1 主流浏览器分布图

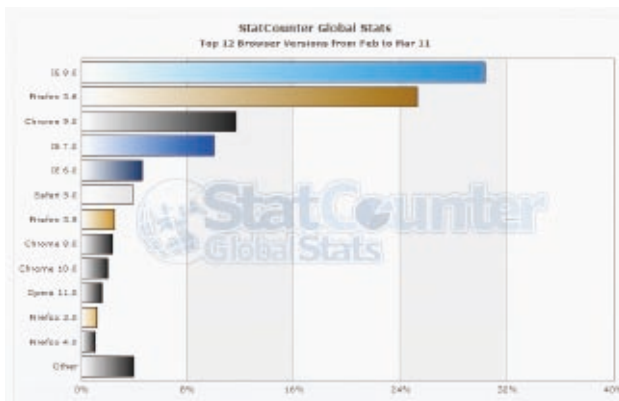


图2 浏览器主流版本分布图

其后，其开放的体系结构也吸引了众多优秀插件，并成为Web开发人员的主要开发调试平台。Google主推的Chrome后来居上，占据了第三名的位置。

浏览器主流版本分布情况如图2所示。

再加上部分最新浏览器版本，我们选取的全部浏览器（版本）如表1所示。

表1 测试浏览器

浏览器（版本）	StatCounter市场占有率统计（%）
Internet Explorer 8	30.34
Internet Explorer 9	0.51
Firefox 3.6	25.27
Chrome 10.0	1.96
Safari 5.0	3.82
Opera 11.0	1.5

HTML5兼容性分析

视频播放标记video

在HTML5技术出现之前，大部分Web视频播放是通过浏览器插件如Adobe Flash来实现，这要求客户在观看视频之前安装相应的组件。video标记的出现使开发人员不再依赖于特定第三方技术，下面的代码片段播放了一个视频文件，设定了画面的尺寸，并带有控制按钮（暂停/进度条等）：

```
<video width="360" height="280"
controls=" controls" >
  <source src="[video url].mp4"
type="video/mp4">
  <source src="[video url].ogg"
type="video/ogg">
  <source src="[video url].webm"
type="video/webm">
```

抱歉，您的浏览器不支持video标记，请尝试其他浏览器。

```
</video>
```

video元素潜在地支持多种视频格式，包括：

- Ogg——采用Theora视频编码和Vorbis音频编码的Ogg视频文件；
- MPEG4——采用H.264视频编码和AAC音频编码的MPEG 4视频文件；
- WebM——采用VP8视频编码和Vorbis音频编码的WebM视频文件。

这些视频格式在HTML5最终标准的接纳程度目前还在各方博弈之中，不过浏览器已经开

表2 视频标记video测试结果

视频格式浏览器（版本）	Ogg	MPEG4	WebM
Internet Explorer 8	×	×	×
Internet Explorer 9	×	√	×
Firefox 3.6	√	×	×
Chrome 10.0	√	×	√
Safari 5.0	×	√	×
Opera 11.0	√	×	√

始选择性地支持Video标记了。表2是兼容性测试结果。

这里有几点需要注意：

- 最新发布的IE9默认只支持MPEG4（H.264编码）格式文件（支持硬件加速），如果安装了WebM/VP8插件，可以支持WebM视频格式。

- Safari浏览器依靠QuickTime做视频播放，所以Safari支持的视频格式与QuickTime一致。请注意：QuickTime在Mac机上是预装的，其默认支持的视频格式是MPEG4，但在Windows系统上QuickTime需要手动安装，也就是说在Windows上默认情况下Safari是不支持所有视频格式的，这一点需要Web开发人员了解。

- Firefox 4.0还将支持WebM格式。

- Chrome已经宣布不再支持MPEG格式的视频。

- Video元素有一个preload属性，可能的值包括：

- Auto——当页面加载后载入整个视频；
- Meta——当页面加载后只载入元数据；
- None——当页面加载后不载入视频。

Web开发人员应该根据网页的实际环境来选择合适的预加载属性以达到较好的前端性能。

音频播放标记audio

与视频标记video类似，音频播放标记audio也是HTML5标准引入的元素，下面的代码播放了一段音频，带有控制按钮：

```
<audio controls="controls">
  <source src="[audio url].ogg"
type="audio/ogg">
  <source src="[audio url].mp3"
type="audio/mpeg">
  <source src="[audio url].wav"
type="audio/wav">
```

抱歉，您的浏览器不支持video标记，请尝试其他浏览器。

```
</audio>
```

audio潜在地支持多种音频格式，包括Ogg vorbis、MP3、AAC、wav等，不同浏览器对audio标记的支持程度表3所示。

表3 音频标记audio测试结果

音频格式浏览器（版本）	Ogg vorbis	MP3	AAC	WAV
IE 8	×	×	×	×
IE 9	×	√	√	×
Firefox 3.6	√	×	×	√
Chrome 10.0	√	√	√	√
Safari 5.0	×	√	√	√
Opera 11.0	√	×	×	√

有几点需要说明：

- 因为存在包含关系，所以音频标记audio的情况基本与video的结果保持一致；

- IE8对video和audio标记都不支持，IE9刚刚引入这些元素；

- audio元素同样具有preload属性，注意正确设置属性值。

绘图标记canvas

在Web 2.0的世界里，炫目的图形是必不可少的组成部分。HTML5引入的canvas元素支持Web开发人员通过JavaScript在页面上绘制线条、图形、添加文字和图像等。下面的代码片段在页面上简单绘制了一个蓝色矩形：

```
<canvas id="test" width="200"
height="100" >
    抱歉，您的浏览器不支持canvas标记，请尝试
    其他浏览器。
</canvas>
<script type="text/javascript">
    var c=document.getElementById("test");
    var cxt=c.getContext("2d");
    cxt.fillStyle="#0000FF";
    cxt.fillRect(0,0,200,100);
</script>
```

Canvas的测试结果如表4所示。

表4 绘图标记canvas测试结果

浏览器（版本）	是否支持canvas标记
IE 8	×
IE 9	√
Firefox 3.6	√
Chrome 10.0	√
Safari 5.0	√
Opera 11.0	√

我们欣喜地看到，各种浏览器对canvas的支持度非常高，对于IE8的限制，可以通过开源项目ExplorerCanvas来解决，只需要两步操作：

- 包含js文件

```
<head> <!--[if IE]><script src="excanvas.
js"></script><![endif]></head>
```

- 直接包含canvas元素或者动态创建它：

```
var el = document.createElement('canvas');
G_vmlCanvasManager.initElement(el);
var ctx = el.getContext('2d');
```

Form 2.0表单元素

对于企业级Web应用来说，表单控件是最重要的页面元素之一。在HTML 5之前，各种类型的表单只能由开发人员通过复杂的属性设置和限制条件（通过脚本计算）来完成。制定中的HTML 5标准引入了一系列分类清晰、功能完善的表单控件标记，包括email、url、

number、range、search、color等，还有表单属性autocomplete、autofocus等。这里提供一个简单例子，用户填写电子邮箱（自动检测格式正确与否）和个人主页（自动检测格式正确与否）等，然后提交给后台系统：

```
<form action="test.jsp"
method="get">
    E-mail:<input type="email"
name="myemail"/><br />
    Homepage:<input type="url"
name="myurl"/>
    <input type="submit"/>
</form>
```

由于HTML5引入的表单控件、属性比较多，这里选取了部分进行了测试，如表5所示。

表5 部分表单元素测试结果

表单元素/属性 浏览器（版本）	email	url	range	autocomplete （属性）
IE 8	×	×	×	×
IE 9	×	×	×	×
Firefox 3.6	×	×	×	×
Chrome 10.0	√	√	√	×
Safari 5.0	√	√	√	×
Opera 11.0	√	√	√	√

几点说明：

- Firefox 4.0将开始支持email、url等表单元素；

- 自动填充已经作为许多浏览器的一项基本功能存在，与HTML 5标准的autocomplete属性需要区别开。

本地客户端存储

Web开发人员经常通过cookie管理客户信息，但是当数据量比较大时，这种方法相对低效，一方面是因为cookie存在大小限制，另一方面每次都通过网络请求来传递。HTML 5引入两种新的存储方式：

- localStorage——没有时间限制的数据存储

- sessionStorage——针对session的数据存储

下面的例子分别使用LocalStorage和SessionStorage实现了简单计数器，结果如表6所示。

```
<script type="text/javascript">
    if (sessionStorage.sessioncount){
        sessionStorage.sessioncount=Number(
sessionStorage.sessioncount) +1;
    }
    else {
        sessionStorage.sessioncount=1;
    }
    document.write("这是session存储里的计数：
"+ sessionStorage.sessioncount+"，重启浏览
器，本计数器会清零！<br>");
    if (localStorage.localcount){
        localStorage.localcount=Number(local-
Storage.localcount) +1;
    }
}
```

```

else{
    localStorage.localcount=1;
}
document.write("这是local存储里的计数: "+ localStorage.localcount + ", 重启浏览器计数器继续计数!");
</script>

```

表6 本地客户端存储测试结果

浏览器 (版本)	localStorage	sessionStorage
IE 8	√	√
IE 9	√	√
Firefox 3.6	√	√
Chrome 10.0	√	√
Safari 5.0	√	√
Opera 11.0	√	√

这项功能对Web开发人员来说非常强大，我们来看一下浏览器对sessionStorage和所有测试的浏览器都实现了HTML5的本地客户端存储标记！不过，这里要提一下安全性，Web开发人员在使用这些元素时，要时刻谨记存储在客户端的数据可能会被授权使用浏览器的其他人查看甚至修改，所以需要注意保存敏感信息。

结束语

本文介绍了一些主流浏览器对部分HTML5技术的兼容性，一言以蔽之——参差不齐，

Web开发人员在尝试采用HTML5技术时，一方面会为其强大的表现力而激动，另一方面也会因为各浏览器的兼容性测试而烦恼，希望本文能够让大家对几个关键技术的支持程度有基本的了解。P

参考文献

1. <http://www.w3schools.com/html5/>
2. <http://www.html5test.com/results.html>
3. <https://github.com/NielsLeenheer/html5test>
4. <http://www.modernizr.com/>
5. <http://modernizr.github.com/Modernizr/test/index.html>
6. <http://gs.statcounter.com/>
7. <http://www.whatwg.org/specs/web-apps/current-work/multipage/index.html#contents>
8. <http://diveintohtml5.org/>
9. <http://tools.google.com/dlpage/webmmf>
10. <http://excanvas.sourceforge.net/>
11. <http://msdn.microsoft.com/en-us/ie/ff468705.aspx>

沃尔玛（中国）投资有限公司

Walmart 中国

诚聘英才加盟（上海、深圳）

To support our growth we are seeking talented IT professionals to join our team.

上海 (Interested parties, please contact : dlee104@wal-mart.com)

- Project Manager-GM eCommerce
- Sr. Java Engineer/Engineer
- Automation QA Engineer
- Sr. Net Engineer/Engineer
- Sr. Software Engineer(QA)
- Sr. Site Performance Engineer
- Sr. Analyst/Analyst
- Site Performance Engineer

深圳 (Interested parties, please contact : isdzp@wal-mart.com)

- Project Manager-Supply Chain
- Workstation Specialist
- Project Manager-BI Area
- Sr. Analyst-Tactical Solutions
- Sr. Engineer-Infrastructure Project Manager
- Infrastructure Engineer Store Setup

More information and positions, please visit <http://www.wal-martchina.com/career/career.htm>

激动人心的HTML5之美

■ 文 / 蒋宇捷

从HTML4诞生以来，整个互联网环境、硬件环境都发生了翻天覆地的变化，开发者期望标准统一、用户渴望更好体验的呼声越来越高。二十年磨一剑，HTML5作为下一代Web标准，它的新特性正在每个新式浏览器的版本中快速地得到支持和体现。随着HTML5和CSS3的发展和完善，Web应用程序正在逐渐表现出桌面应用的性能和功能，和桌面应用之间的区别将越来越模糊。未来不能用HTML5来实现的功能将越来越少，未来绘图、视频编辑、3D建模等也将从单机软件转到浏览器上通过Web应用程序来实现。

下文我将结合实例，谈谈激动人心的HTML5之美。

语义之美

语义化很美，在合适的地方使用合适的标签，把人和机器一视同仁。书写语义化的页面就像建造符合工业标准的建筑，阅读语义化的页面像阅读一本你熟识的优美著作。

HTML5的语义化标签包括：

<section>-代表文档中的一段或者一节；

<nav>-用于构建导航；

<article>-表示文档、页面、应用程序或网站中一体化的内容；

<aside>-代表与页面内容相关、有别于主要内容的一部分；

<hgroup>-代表段或者节的标题；

<header>-页面的页眉；

<footer>-页面的页脚；

<time>-表示日期和时间；

<mark>-文档中需要突出的文字。

和采用css+div进行页面布局的方式相比，这些新标签明确的表现了页面元素的结构和含义。下面的代码展示了这样的一个示例：

```
<!DOCTYPE html>
<html>
<head>
  <meta charset=" utf8" >
  <title>HTML5</title>
  <link rel=" stylesheet" href=" html5.
css" >
</head>
<header>
  <h1>Title</h1>
  <h2>Subtitle</h2>
</header>
<aside>
  <nav>
    <h2>Nav1</h2>
    <ul>
      <li>Link1</li>
      <li>Link2</li>
    </ul>
  </nav>
  <nav>
    <h2>Nav2</h2>
    <ul>
      <li>Link3</li>
      <li>Link4</li>
    </ul>
  </nav>
</aside>
<article>
  <header>
    <hgroup>
      <h1>HTML5 is beautiful</h1>
      <h2>Semantic, Natural, Simple
and Useful</h2>
    </hgroup>
```



作者简介：

蒋宇捷，腾讯研究院浏览器前端负责人，曾担任傲游Web后台主管，西安交通大学硕士。HTML5研究小组成员，2008年、2009年Perl大会演讲者。关注Web前后端架构、移动互联网应用和团队管理。

```

</header>
<section>
  <h2>Semantic</h2>
  <p>HTML5 is Semantic.</p>
</section>
<section>
  <h2>Natural</h2>
  <p>HTML5 is Natural.</p>
</section>
</article>
<footer>
  <a href="http://www.w3.org/TR">W3C</a>
</footer>
</html>

```

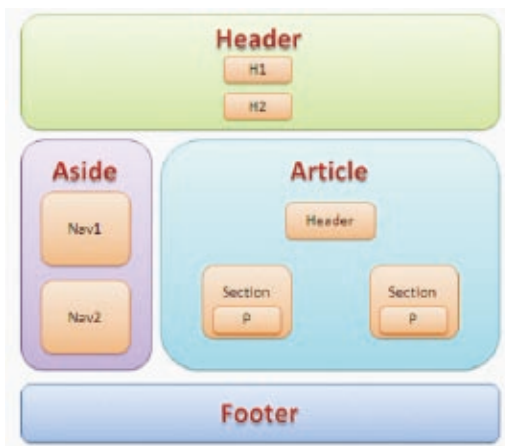


图1 页面元素结构图

相当简单和清晰，我可以不添加任何注释，人和机器都明白它所要构建的结构和内容。

目前主要浏览器的新版本都支持HTML5语义化标签。

人性之美

HTML5设计理念中的通用访问体现了人性化之美。通用访问包括可访问性、媒体中立和国际化支持三个概念。可访问性考虑了对残障用户的支持，媒体中立为所有的平台和终端上（例如Android和iPhone的平台上）建立统一标准；而国际化的支持体现在不同的语言和书写习惯上。

就像CSS3中对国际化的支持体现在padding-start, padding-end（在Chrome新版本的设置中心中可以看到它的应用）等新属性一样，HTML5引入了Ruby标签。Ruby标签允许为一个或者多个文本添加附加注释（其命名来源于Ruby字符），例如下面的代码：

```

<p>Hello,
  <ruby>
    <rb>HTML5</rb>

```

```

    <rp> (</rp>
    <rt>Hyper Text Mark-up Language
  </rt>
  <rp>) </rp>
</ruby>
</p>

```

显示的文本如下：

Hello, Hyper Text Mark-up Language 5
HTML5

Ruby标签在中文、韩文和日文教科书和古文中非常有用，读者们可以方便了解文本的详细读音和含义。下面的诗句很美，而HTML5更美。

死生 契^{qì} 阔^{kuò}，与子成说。执子之手，与子偕老。

有趣的是在IE、Firefox或Opera中复制粘贴上面的文本到编辑器中，会显示下面的文本，这是因为<rp>标签的特殊作用：允许在不支持Ruby标签的浏览器中优雅的降级，同时支持非格式化的复制和粘贴。

死生 契（读音：qì）阔（读音：kuò），与子成说。执子之手，与子偕老。

目前主要浏览器的新版本对于Ruby标签的支持程度如下：除Firefox和Opera外均提供支持。

简单之美

大道至简，化繁为简是许多标准、规范、框架的终极目标。HTML5很好地诠释了这一点，例如以浏览器原生能力支持和代替复杂的Javascript代码、提供简单而强大的新API。

HTML5自身即跨浏览器的JS库，使得开发者的工作更容易：例如新的表单标签和属性、内置拖拽事件使得完全不需要Javascript的支持就能实现许多常用的功能。

我们再来看HTML5表单，它实现了许多简单易用的原生属性和控件，可以轻松的构建强大的表单，帮我们节约大量的Javascript代码。下面是一个HTML5表单的示例：

```

<form>
  <p>
    <label for="email">电子邮箱: </label>
    <input type="email"required autofocus name="email" id="email" placeholder="您的电子邮箱">
  </p>
  <p>
    <label for="username">用户名: </label>

```



```

        <input type="text" pattern
        ="\w{6,12}$" required name="username"
        placeholder="6到12位的英文字符">
        <input type="search" placeholder="
用户名搜索" autosave="www.yujie.com"
results="5" name="username-search"
id="username-search">
    </p>
    <p>
        <label for="username-search">生
        日: </label>
        <input type="date" min="1980-
        01-01" max="2011-3-16" name="birthday"
        id="birthday" value="1982-10-10">
    </p>
    <p>
        <label for="blog">博客地址: </
        label>
        <input type="url" name="blog"
        placeholder="您的博客地址" id="blog">
    </p>
    <p>
        <label for="mobile">手机: </label>
        <input type="number" name="mobile"
        pattern="^[0-9]{10}$" id="mobile"
        placeholder="您的手机号">
    </p>
    <p>
        <label id="label-working-year"
        for="working-year">工作年限: </label>
        <input type="range" min="1"
        step="1" max="20" name="slider"
        name="working-year" id="working-year"
        placeholder="您的工作年限" value="3">
    </p>
    <p>
        <label for="age">年龄: </label>
        <input type="number" name="age"
        id="age" value="20" autocomplete="off"
        placeholder="您的年龄">
    </p>
    <p>
        <label for="avatar">头像: </
        label>
        <input type="image" src="user.
        png" name="avatar" id="avatar"
        placeholder="点击选择头像">
    </p>
</form>

```

图2 HTML5表单

示例中包含了email、search、date、url、range、number、image类型的输入框和required、autofocus、placeholder、pattern、autosave、results、min、max、step、autocomplete等新属性，它们简单到根据名字就可以揣测出用途。效果如图2所示。

HTML5表单在各浏览器里的支持程度和表现并不一致，例如Search输入框目前只被Webkit内核浏览器支持，而日期弹出框和拖动条刻度仅被Opera支持。以上的标签类型和属性目前没有一个浏览器完美支持。

实用之美

HTML5的Web Worker、Web Socket、Web Storage等新API让很多后台的工作可以放到前端来处理，Web Worker解决Javascript单线程和阻塞的问题，相当于提供了分布式处理的框架；Web Socket提供了全双工的长连接通信，利用它，我们可以实现微博消息推送、新邮件推送、实时游戏和聊天，减少了不必要的数据传输，提高了信息的实时性；Web Storage相当于前端的Memcached和数据库。

而HTML5的Canvas是最强大的API之一，可以动态生成图形、图像和动画，在HTML5游戏中使用的非常普遍。在下面的例子中，我将展示HTML5 Canvas之美：实现图像颜色渐变效果。

以前，我们在网站上显示灰度图像到彩色图像的渐变动画有两种实现方式：1、基于IE滤镜的方案，缺点是无法实现浏览器兼容；2、手动创建彩色图像的灰度版本。现在，利用HTML5的Canvas画布，我们可以高效简单的实现此动画效果。



图3 利用canvas实现动画效果

将鼠标放在左边的图片上，图片颜色将从灰度渐变到彩色，一切就在眼前鲜活起来。

核心的Javascript代码如下：

```

// 加载时就进行处理
$(window).load(function(){
    var img = $('#color-img');
    // 复制图像
    img.clone().addClass('gray-
img').css({ "position": "absolute",
    "z-index": "2", "opacity": "0" }).
insertBefore(img);
    img.attr('src', grayscale(img.
attr('src')));
    // 图像的淡入
    $('#color-img').mouseover(function()
    {
        $(this).stop().animate({opacity:
1},1000);
    })
    // 图像的淡出
    $('#gray-img').mouseout(function(){
        $(this).stop().animate({opacity: 0},
1000);});
    });
    // 创建灰度版的图像
    function grayscale(src) {
        // 取得canvas元素及其绘图上下文
        var canvas = document
createElement( 'canvas' );
        var ctx = canvas.getContext( '2d' );
        var imgObj = new Image();
        imgObj.src = src;
        canvas.width = imgObj.width;
        canvas.height = imgObj.height;
        ctx.drawImage(imgObj, 0, 0);
        // 绘制一副图像
        var imgPixels = ctx.getImageData(0, 0,
canvas.width, canvas.height);
        // 获取之前的数据
        for(var x = 0; x < imgPixels.height;
x++){
            for(var y = 0; y < imgPixels.
width; y++){
                var i = (x * 4) * imgPixels.width + y
* 4; var avg = (imgPixels.data[i] +
imgPixels.data[i + 1] + imgPixels.data[i
+ 2]) / 3;
                // 计算灰度值

imgPixels.data[i] = avg;
                // rgb中的r
imgPixels.data[i + 1] = avg;
                // rgb中的g
imgPixels.data[i + 2] = avg;
                // rgb中的b
                // i + 3是alpha通道, 我们现在不需要
            }
        }
        ctx.putImageData(imgPixels, 0, 0, 0,
0, imgPixels.width, imgPixels.height);
        return canvas.toDataURL();
    }
}

```

针对不支持的旧浏览器，我们可以使用Modernizr Javascript库或者原生Javascript检测当前浏览器是否支持，并提供替代性的解决方案：

```

if(!Modernizr.canvas) { // 或者使
用!document.createElement( 'canvas' ).
getContext
    $(document).ready(backupFunc);
}

```

目前主要浏览器的新版本都提供Canvas标

签的支持。

HTML5的不足

目前HTML5还有哪些不足之处？

安全：Firefox 4的WebSocket和透明代理的实现存在严重安全问题，同时WebStorage、Web Socket这样的功能很容易被黑客利用，来盗取用户的信息和资料，另外HTTP的机制导致了Web应用安全性有所欠缺，这将在很长的时间内成为问题。

性能：某些平台上的引擎问题导致HTML5性能低下。同时在不加入GPU加速的情况下，HTML5处理复杂音视频、动画的性能不尽如人意。


完善性：HTML5还在成熟和发展中，像之前曾经支持的Web SQL Database，W3C已经决定不再维护。各浏览器对许多特性的支持程度也不一样，而地理定位API在国内的前途还不清晰。

技术门槛：HTML5简化开发者工作的同时代表了有许多新的属性和API需要开发者学习，像Web Worker、Web Socket、Web Storage等新特性要求对于后台的技术需要有一定的了解，甚至需要深入了解其后面原理和逻辑，而Canvas要求视觉和图像算法的一些知识。传统的前端开发者需要掌握更多算法、视觉、后台甚至浏览器原理的知识，机遇的同时也是巨大挑战。

总结

上面的大多数不足是HTML5发展过程的中间状态导致的，从长期来看，浏览器厂商的支持、移动互联网的蓬勃发展使得HTML5的前景一片光明，开发者对此必须要有清楚的认识和长远的目光，提前做好技术储备。

对于HTML5，我认为不仅要了解技术细节，还要从根本上理解HTML5的架构，从更高的层面上理解它的深层次含义，它的设计理念，你会发现它的架构之美，对于个人架构设计能力和境界提升有很大帮助。

Web应用程序和移动互联网是大势所趋，IE6终将消亡，Web和浏览器标准终将统一，HTML5就是道之所藏、美之所在，让我们一起迎接和拥抱HTML5的到来！

清音浊言——献给HTML5

■ 文 / 王子龙

HTML5成为独立名词尚不久远，然其锐不可挡，隐然以星星之火而渐成燎原。作为一名逐波客，在此为诸位献上些许清音浊言。

HTML5已经带来哪些变革？不同的角色、不同的载体、不同的领域又将会受到怎样的影响？本文试图解答你心中的疑惑，希望可以抛砖引玉。

一千个人眼中有一千个哈姆雷特 如果你专注于产品

HTML5正在促使互联网产品的革新。它可以让你的产品获取更多的用户信息，如所持终端的地理属性。它解决了现有产品中的一些技术壁垒，可以延伸出更多的产品方向。

HTML5使得开发重量级应用变得容易（Local Storage、Application Cache、Custom content handlers、FileReader、FileWriter等API）。原本常见的桌面软件如图像绘制/处理（如图1，一个在线服务Darkroom，可以使用Flickr、Picasa等账户登录，在线处理照片）、视频编辑、文档编辑、远程帮助、在线编程（如图2,Cloud9IDE）等，可以考虑提供基于Web的替代品，虽然HTML5的版本功能不如桌面版本那样全能，但能解决当用户急需查看某



图1 Darkroom

种文件时，软件还没有安装在电脑中的尴尬。你所要做的是去熟悉Web服务的特性，以及与传统应用的相异之处。

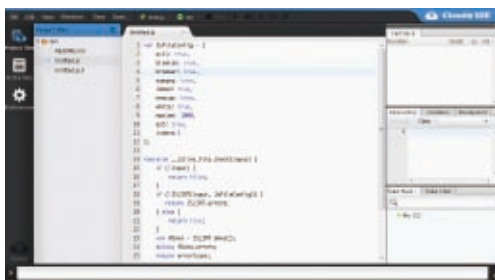


图2 Cloud9 IDE

HTML5很好地解决了实时数据通信的问题，尽管同时也引发了安全方面的考虑。

（WebSocket协议用于解决这类问题，Mozilla、Opera等虽然在浏览器中实现了此技术，但目前由于安全方面的考虑，已将之禁用，直到此协议更新并解决安全隐患）。原本基于同步/更新服务类产品，如多人实时游戏、在线会话，协同工作（如图3，EtherPad）等，可以逐步减少对于Flash或是桌面软件的依赖，转而开发与维护HTML5版本。尽管目前在性能方面HTML5不能完全达到桌面软件的级别，但不要忘记，浏览器在不断加速。

得益于手持终端（手机、平板、部分电子书）的普及与其对于HTML5的良好支持，基于地理位置的服务（LBS）正蓬勃发展，你是否

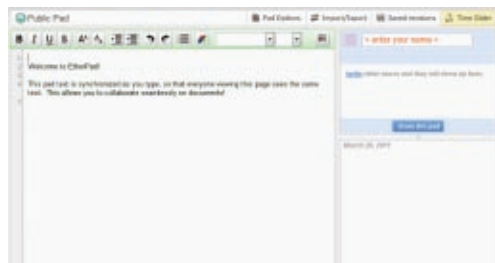


图3 EtherPad



作者简介：

王子龙，2010年毕业于南京大学，同年加入百度，任职于百度电子商务事业部前端开发团队WED（We enjoy days）。

也想分一杯羹?

HTML5现已大幅增强对于图形图像的处理能力(基于Canvas、WebGL)。轻量级的社交游戏在手持终端上也能良好运行,从事游戏设计的你是否也推出掌上版本?现有产品是Impact,国内创新工场也将提供中间产品。

HTML5草案中已标准化跨域通讯方式,意味着不同来源的产品可以通信与合作。HTML标准的前进也将促使搜索引擎改变其排序规则,这导致你需要考虑重新优化产品。

如果你有空闲时间,那么去了解HTML5的发展历程,了解标准的建立与推广背后那些大公司的角逐,了解它与Flash的是是非非,了解它最根本的设计原理,将是很大的收获。

如果你专注于UE

HTML5的到来让你可以大展拳脚。CSS3(严格来说CSS规范并不属于HTML5)提供了更多的表现方式,如渐变、圆角、阴影、字体、布局等属性(border-radius、box-shadow、text-shadow、border-image、gradient,如图4所示按钮),使得前端开发人员更容易去实现你的设计稿,也意味着可以更加自由地设计。

追求完美的你可以给予用户最好的互动,

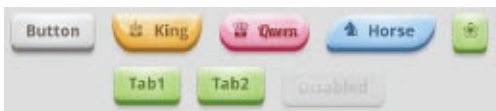


图4 由CSS3实现的按钮

比如使用动画作为操作反馈,设计键盘、鼠标、触摸多种交互方式,提供近似系统原生的操作方式(如拖拽),提供即时信息提示。在此之前,你需要向开发人员了解哪些操作,你的产品受众的客户端支持;了解哪些交互表达方式,是开发人员可以实现的;了解在浏览器实际操作层面,网页产品怎样表现更好。

HTML5给你带来的额外工作还有:你需要设计不同的尺寸、排版、内容,来应对不同的展现环境。手机、平板、PC,一个也不能少。尽管增加了工作量,但无疑会带来更好的体验与更多用户。

如果你专注于前端

如果HTML5可以真正普及,那么对于前端开发人员来说将是福音。尽管前端开发人员需要真正准确、仔细地去了解HTML5的概念与规范;花费大量精力学习其技术细节;而且会

面临更多向前兼容、安全隐私问题,但毋庸置疑,HTML5将为你、为大家解决很多麻烦,它试图让你更轻松的编写代码、实现产品。

就以曾经使得很多前端开发人员焦头烂额的表单为例,相比于HTML4.01,HTML5额外制定多种表单控件规范(<http://www.w3.org/TR/html5/forms.html>),如搜索框,URL输入、邮件输入、日期选择、颜色选择等(如图5),HTML5规范详细说明这些控件的交互方式,再加上required、placeholder、min、max等属性,以及表单验证、多文件上传等功能,这些都意味着你可以逐渐减少对于庞大的第三方JavaScript UI控件的依赖,可以无需纠结于衍生出的不同组件的设计/体验差异,可以轻松应用于手持设备!重要的是这样的好处显而易见:减少流量、简化代码、易于维护、统一使用体验。



图5 表单

HTML5语法简洁、语义明显,并且向前兼容,同时新增元素标签在处理文档结构、页面UI、排版/布局方面堪当大用(Section、Semantics)并能有效简化你的代码。另一方面,CSS3提供完备的CSS选择符(IE9已经能支持大部分的CSS3选择符)也将使你能更惬意的实现UE的设计,逐步抛弃原本各式各样额外的解决方案。

Canvas元素将是你的试验田,它可以用来替代部分原来Flash专属的技术。使用它强大的绘图能力可以制作动画、绘制图表、图像处理以及提供游戏绘图支持,见图6~9。不仅仅是2D,结合WebGL,3D游戏也不再话下。目前Canvas画布绘制效率在不依赖硬件加速的情况下,无法达到Flash的程度。需要巨量重绘/计算的代码的产品还是有Flash实现较好。同时尽管未来的发展方向之一是提供完备API去获取本地硬件与用户数据,如摄像头、麦克风等设备以及账户联系人、通信记录、日程安排等信息,但目前实现此类应用仍需使用Flash,浏览器在此方面尚不完

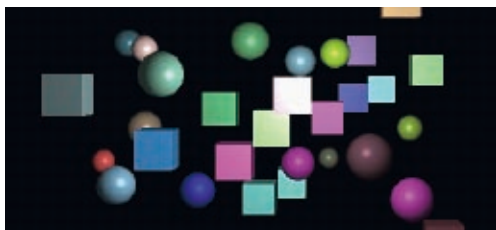


图6 Canvas绘制示例，基于WebGL

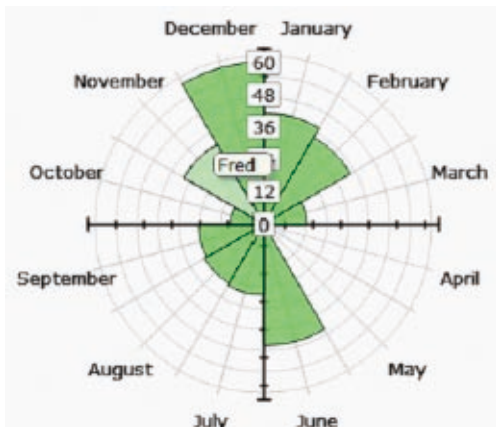


图7 Canvas绘制图表，Rgraph



图8 数据可视化，ProcessingJS



图9 象棋游戏

备（并非不能，而是标准尚未统一）。

Video标签的出现同样可以减少你的部分工作，或许将逐渐替代Flash播放器，尽管目前媒体编码格式尚未统一，但在背后角逐的大公司总会慢慢达成一致。这意味着在网页中嵌入视频如同写一个img标签一样简单，当然代码会多几行，此外更为吸引你的是，完全可以通过CSS去定义那些Control的样式。

Application Cache可以支持此特性的浏览器中有效减少网站流量。WebStorage使得本地存储成为可能，基于Flash、Google Gears

（Google已经停止Gears的继续开发与更新）的本地存储代码将会成为过去式。WebSocket协议提供长连接的实现方式，无需依赖Flash。同时HTML5标准里还有Database的实现（Web SQL Database提案已被IndexedDB取代，但是Webkit等浏览器引擎仍然提供其实现），这些都为开发重量级应用做好了准备。另外还有Drag and drop之流，尽管已经不再新颖，但Gmail本地文件拖拽长传功能一度为大家津津乐道。其他诸如Session history、iframe方面较HTML4.01做出的标准修改以及尚未被浏览器实现的Microdata，你也可以深入了解其具体原理。

HTML5使得你具有跨平台开发产品的能力，同时为之产生的学习成本远低于学习某个平台的SDK。HTML5设计原理不仅对于产品设计人员有益，对于前端开发人员来说，也是不可多得的饭后甜点。最后给先辈们与同侪从事前端之后的个人感触：不断学习以进步。

天下谁人不识君 我是一台老式PC

我运行着已经10岁的Windows XP，那么我的浏览器可能叫做因特网探险家6（IE6），或者是它的兄弟，代号7。HTML5对于这些家伙来说有点陌生，4.01才是他们的最爱。（IE6~IE8的高比重仍旧是国内现况）。

我是台新本，强劲稳健

Windows 7、Mac我喜欢，IE9、Firefox4、Chrome 10、Safari 5站左右，这是可见的未来。HTML5与同一阵营，它从头到脚我都了解（IE9对于HTML5的支持不如其他几位）。

我是智能手机

iPhone、Android是我的代名词，也有人称呼我BlackBerry、Window Phone。虽然我的浏览器很智能，可是我才10岁未成年，HTML5的一些零件对于我来说有点重（目前手机端浏览器性能仍然无法直接和PC媲美，尽管最近Safari再次更新），但你不要着急，我很快会长大。

我是平板设备

我比我兄弟智能手机强壮一点儿，已是同学少年。HTML5我很喜欢，我曾经的教主乔给我说过它的好处。

我花开后百花杀 站在游领域

HTML5对于游戏厂商未尝不是一个契机，无论是休闲社交游戏，还是大型游戏都可以在不同设备上大展拳脚。由此衍生出的游戏中间

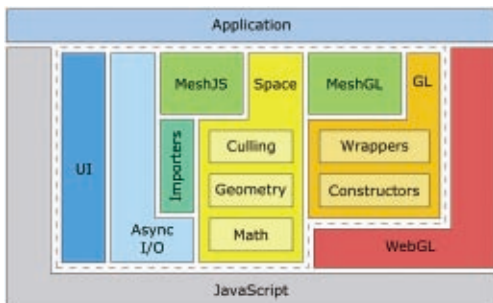


图10 SpiderGL结构（其中Mesh部分为3D模型定义解析与渲染单元。Space部分为数学/几何计算单元）

件也将逐步成熟，比如SpiderGL（图10），基于WebGL的中间件。

国外已有基于HTML5的大型游戏运行于浏览器的案例Quake II GWT Port（图11）。Quake II使用了什么呢？WebGL 3D、Canvas、Audio、WebSockets、LocalStorage。WebGL用于渲染，WebSockets使得多人即时游戏可行，需要的工作为从原本的UDP接口转为此接口。使用LocalStorage模拟本地文件系统，用于读写游戏数据与配置。



图11 Quake II GWT Port 运行于浏览器

桌面应用领域

HTML5草案的发展方向之一就是对于桌面级应用的支持，比如草案中，已经有部分浏览器支持的特性：运行`window.navigator.registerProtocolHandler("remem", "https://remxxx.info/?uri=%s", "Rem2015")`；这样的代码将会出现如图12。

当用户下次点击以`remem://`这样类型协议的链接时，则会跳出图13。

传统的应用以及文件处理将有可能逐步转



图13 文件/url打开方式

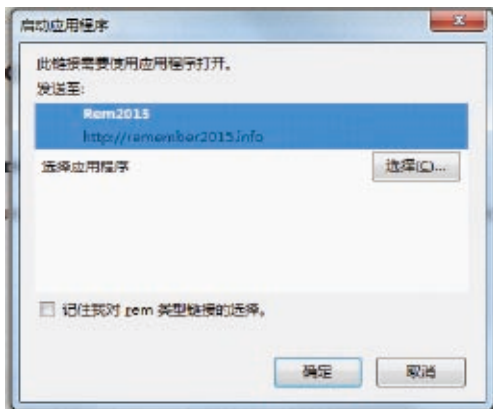


图12 添加应用程序处理

向网络服务。可能由不同软件/服务提供商为传统桌面文件建立各种协议，统一不同文件类型的打开方式。比如微软即可注册Doc、PPT等文件的在线处理方式Office Web Apps。桌面软件通过提供网络处理与存储服务将逐步完善甚至替代原本桌面版本的功能。越发强劲的浏览器，跑得更快的JavaScript引擎，为客户端的大型应用提供了无限可能。

站在传媒、信息领域

Video使得视频的运用大大简化，虽然目前浏览器对于视频格式的支持尚未完全统一，但不同浏览器给予用户的视频观赏体验基本一致，很好地降低终端用户的学习成本。此外诸如字幕等特性虽然尚未有浏览器支持，但是已经加入草案。

2010年美国电子期刊已经超越传统期刊。随着电子书、手持设备中个性化的内容阅读器的流行，信息聚合服务需要应对更多的平台，而HTML5是一个不错的选择，它提供不逊色于纸张的内容展现以及更好的阅读体验。

网络与终端正在为它们的使用者解决越来越多的问题与挑战，随着越来越多的产品问世，网络这个大平台给予的体验正在逐步超越原生操作系统；从基本的图形图像、信息聚合到地理信息服务、云端计算/存储都在默默改变大家使用计算机和看待网络的方式。随着网速的激增，本地与网络的差距将会原来越小，界限将会越来越模糊。让我们拭目以待。P

HTML5 2D游戏开发实例剖析

■ 文 / 魏子钧

离HTML5规范正式发布还有一段时间，但是随着规范本身的日趋完善，以及各大浏览器的集体发力，HTML5技术离我们越来越近。在诸多HTML5特性当中，Canvas是被支持度最广，最吸引眼球的一个。利用Canvas，可以在Web页面中轻松实现图形图像的绘制，进而实现动画，甚至游戏开发。

本文将通过简单的实例，向大家介绍如何用HTML5 Canvas技术开发2D游戏。



该游戏的画面和玩法都极其简单：按键A/D控制马里奥左右移动，W控制马里奥跳跃。目的是尽量躲避下面来回移动的敌人（乌龟），一旦碰到就结束游戏。这款游戏极其粗糙和无聊，但通过它可以了解游戏开发的大致流程。简单示例更容易让初学者最快收获一份小小的成就感，这通常是驱动我们更进一步的原动力。

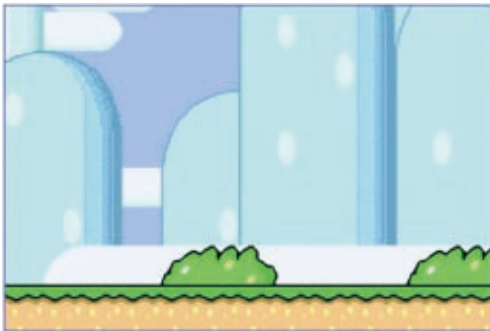
静态图片的绘制

开发游戏之前，先了解如何利用HTML5绘制静态图片，这是开发2D游戏的基础。在HTML5 Canvas中实现绘图非常简单，大致流程如下：

- 创建一个Canvas，并设置大小
- 取得Canvas的Context

- 加载图片
- 通过Context绘制图片

通过运行实现代码，我们可以在Canvas上绘制完整的bg.png。



这其中主要涉及了3个Canvas相关的API，简单说明一下：

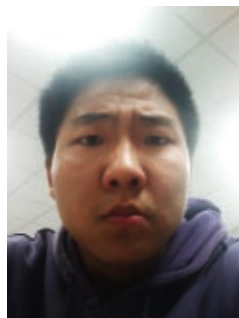
- `Canvas.getContext(“2d”)` 取得2D类型的绘图上下文，可以把它理解成我们和Canvas之间的中介，基本上绘图操作都是通过Context来完成的。

- `Context.drawImage(img,dx,dy)` 将图片绘制在画布的[dx,dy]坐标处，以图片的左上角为基准。

- `Context.drawImage (img,sx, sy, sw, sh, dx, dy, dw, dh)`将图片的一部分绘制在画布上。其中[sx, sy, sw, sh]定义了源图片中的一个矩形区域：左上角坐标为sx,sy，宽高分别是[sw,sh]，[dx,dy]为要绘制的目的坐标（绘制在画布的什么位置上），[dw,dh]为绘制的目的大小。通常我们不需要进行任何缩放，所以dw==sw，dh==sh。

上面的例子只是加载并绘制了一张图片。而开发游戏时，我们往往会需要多张图片。

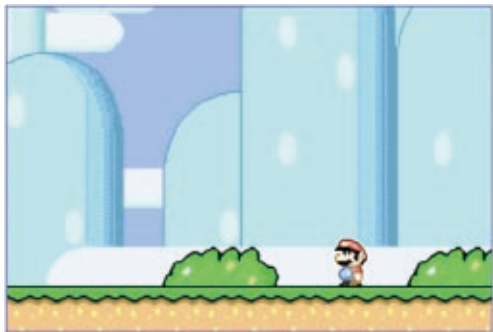
为了更好地加载并管理这些图片，通常需要编写一个简单的资源加载的工具函数。



作者简介：

魏子钧：Web开发工程师，并拥有多年JEE研发经验。Javaeye社区资深会员。目前专注于HTML5游戏引擎的开发，以及HTML5技术在国内的推广。现就职于惠普（上海）PSG 部门。

运行工具函数后我们可以看到，画布上先画了一张完整的bg.png图片。



然后又会在画布坐标 [200,100]处绘制了player.png 这张图片中的一小块区域（左图蓝框内的区域）。

大家会发现，在Canvas上绘制的图片，并不会抹掉之前绘制的内容，只会部分覆盖。如果我们想抹掉之前的内容（清空画布），可以使用 context的另一个API：clearRect。

context.clearRect(x,y,w,h)清空画布上的一块矩形区域。清空整张画布可使用 context.clearRect(0,0,canvas.width,canvas.height)。

虽然Canvas（准确的说应该是Canvas的Context）提供了很多API，但是目前我们掌握的这两个drawImage和clearRect已经足够我们来开发一个简单的2D游戏。

让图片动起来

介绍了如何在Canvas上绘制静态图片，如何让我们绘制的图片动起来呢？

通常一个简单的动画实现流程如下：

1. 初始化图片和画布；
2. 更新图片状态（例如,位置）；
3. 擦除原图片；
4. 绘制新图片；
5. 休眠一会儿；
6. 如果不希望终止动画，跳转到第2步；
7. 结束。

其中第2到第6步，构成了无限循环，通常被称作“动画（或游戏）主循环”，以下简称主循环。

第3步，我们可以使用clearRect清空画布的方法来擦除原图片，如果有完整的背景图片

（不能小于Canvas），也可以使用背景覆盖的方法。当然还有更好的方法，例如脏矩形等，本文暂不讨论。

第5步，休眠时间的单位是毫秒，具体休眠多久取决于整个动画的刷新频率。动画和我们看的视频很类似，主要就是通过不断重绘来实现动态效果的。每秒钟重绘的次数通常叫做FPS（帧频，每秒钟可以绘制的最大帧数）。一段视频FPS为30看起来就已经很流畅了，对于动画和游戏而言，类型不同对FPS的要求不同。

实现动画序列

接下来该让让静态图片变成“有生命力”的动态图片。一组静态图片，按照一定的速率和顺序进行显示，就共同构成了动态画面（Animation）。在游戏开发中，也是用类似的思路来实现动态效果。

Animation中的每一幅图画，我们称作Frame（帧），实际上，Animation可以理解为一个管理器或播放器，它来管理和控制一组Frame的播放。游戏中通常用Animation表现物体的动作姿态，例如向左走，向右走，匍匐，跳跃等。虽然每一帧显示的内容不同，但是通常不会把每一帧对应到一张单独的图片，而是会制作一张较大的“帧的集合”，否则会增加加载图片时的I/O次数，产生大量图片文件，不利于管理。

制作“帧的集合”没有统一标准，可根据图片量的多少以及游戏的需求自行选择，本文采用“一个角色的多个Animation的所有frame都放到一张图片上”这种方式。要实现一个Animation，首先要定义Frame。把若干个这样的frame放在一起，依次进行更新和绘制就可以产生动画。

还有一种播放策略是根据当前Animation整体已经播放的时间，来得到当前应该播放的帧，两种方式各有优缺点。本文暂不考虑该方式。

我们需要写一些函数来初始化并不断播放这个Animation。循环播放Animation的流程如下：

1. 初始化Animation；
2. 将第0帧设置为当前帧；
3. 判断当前帧是否播放完毕，如果已经完毕，则激活下一帧。如果没有下一帧，则再次从第0帧开始；

4. 绘制当前帧；
5. 休眠一会儿；
6. 从第3步开始。

不难发现，这和移动图片时的流程非常相像。为了更好的实现这个流程，也为后面更复杂的场景打下一个坚实的技术，我们可以引入面向对象的思想，对animation相关代码进行重构，封装出Animation类。

运行这个示例可以看到一个原地踏步的马里奥。下面我们结合移动图片的示例，让原地踏步的马里奥真正走动起来。实现Animation 和移动的主循环可以合并成一个。事实上，游戏开发中通常也都是合并的。无论有多少个动画，游戏主线程都只有一个。

游戏中的精灵

我们已经可以看到一个“向右走”的马里奥，但是一个人物往往会有多种状态和动作。同时一个人物也会有很多的属性。

把人物的各种动作和属性，放到一个对象里，进行统一的管理，是个不错的主意。而这个对象，就是我们常说的精灵（Sprite）。下面引入面向对象的思想来编写Sprite类：

Sprite对象可以是游戏里的人物，也可以是道具（门、食物）。把游戏中的哪些物体定义成Sprite，要视游戏而定。在本文示例中，主角和敌人都定义为Sprite。

输入与控制

走来走去的马里奥，不是一款游戏，如何实现玩家对游戏角色的控制呢？对于通过键盘操控的游戏，我们可以通过监听浏览器的keydown/keyup事件来进行检测。

通过代码记录按键状态后，就可以在游戏的主循环里根据不同按键状态来执行不同的操作。

前面，我们通过判断马里奥的位置来改变运动方向和Animation，下面来看一看如何通过键盘来控制马里奥左右移动以及跳跃。左右方向是匀速直线运动，所以的原理和前述的类似，通过改变speedX的正负来改变方向。但是跳跃相对复杂些，跳跃在垂直方向上属于一种上抛运动，要引入起跳初速度和重力加速度。由于处理玩家输入以及改变马里奥运动状态的代码比较多，且有相关性，所以给马里奥的精

灵增加一个handleInput方法。

这个handleInput方法还是比较好理解的，但是何时调用它是一个值得注意的问题。我们通过按键，改变的是移动的速度，而不是直接改变坐标。如果要让改变的速度生效，这个速度必须要持续一段时间。而这一段时间通常就是主循环两次迭代之间的间隔。也就是说，我们第n次迭代时改变的速度，要让它第n+1次生效才有意义。所以我们在迭代的最后调用handleInput方法。

游戏主控类

本文示例中，除了主角马里奥之外，还有5个来回移动的敌人。他们也是精灵。随着精灵的增加和代码的进一步复杂话，我们需要再次重构，封装出游戏总的控制类：Game。Game类中包含多个Sprite，在每次迭代时，由Game负责所有精灵的状态更新和绘制。同时游戏的canvas context、全局的基本属性（如宽高、FPS）、启动函数、主循环等也都包含在Game里。有了这个主控类之后，创建游戏的流程如下：

1. 创建Animation对象；
2. 创建Sprite对象，并加入相应的Animation；
3. 创建Game对象，并把Sprite加入Game；
4. 初始化Game对象，同时初始化内部的Sprite，以及Sprite内的Animation；
5. 开始游戏。

当游戏中产生新的精灵（如出现新敌人），旧的精灵消失（被消灭的敌人）时，只要对Game中的sprites集合进行操作即可。

敌人与碰撞检测

有了主控类，向游戏中添加敌人就变得容易了许多。下面试着向游戏中加入5个敌人，由于敌人的外观和属性基本相同，不同的就是初始的坐标，和移动的速度，所以可以将创建敌人的过程封装成一个函数。在游戏初始化之前，将精灵加入到Game主控类中。

现在运行游戏后，就可以看到一个马里奥和一群来回移动的敌人了。有了主角和敌人，下一步要做的就是对两者进行碰撞检测，判断主角和敌人是否有接触，如果接触到了便结束游戏。对于简单的



2D游戏，通常只需取Sprite当前Frame的外框，作为碰撞区域，进行简单的矩形碰撞检测即可。当上图中红色和蓝色矩形有交集时，则认为马里奥和敌人它们发生了。

现在我们就可以使用`enemy.collideWidthOther(player)`来判断敌人是否和主角发生了碰撞，如果碰撞了就提示“Game Over”。

记录分数

现在，已经完成了游戏核心部分的开发，下面加入对分数的记录和显示，这里的分数就是在游戏中坚持的时间。首先在页面中加入一个简单的浮动div，用来显示分数。

```
<div id="statebar" style="font-size:24px; position:absolute; top:10px; left:280px;" >
Time : <span id="timeCount" ></span>
</div>
```

然后在游戏主循环中，更新这个div的显示。

每次执行到主循环时，就会更新玩家当前的游戏时间。至此，本文开头提到的那个简单的示例就开发完成了，虽然简陋，但是已经可玩儿了，也具备了一些游戏的要素。

小结

本文中的示例并不能叫做一款游戏，它还缺少有很多游戏必备的元素。

如果要把这个示例变成一款真正的游戏，我们还有什么工作要做。

1. 更好的创意、更多关卡、更多挑战，当然，这是一个非技术问题。

2. 加入声音。虽然对于很多游戏来说，声音不是必需的，但是没有声音，游戏的魅力会大打折扣。好在我们有HTML5 Audio来帮我们实现这个功能。

3. 更好的画面表现力。更漂亮的画面和更多的Frame，将会让游戏视觉效果更出色。

4. 更精确的碰撞判断。现在采用Frame所在矩形区域来代表碰撞区域，这显然存在很多问题。如下图，视觉上两者并没有碰撞，但是在游戏中会认为已经碰撞了，因为两者的Frame矩形有交集。要解决这个问题，可以单独定义碰撞区域，而不是简单的使用Frame所在矩形。

5. 游戏UI。这里的UI是指游戏的各种交互

界面，如加载页面、主菜单、暂停界面、游戏结束界面，等等。在HTML5游戏中，我们可以方便地利用HTML中的元素，如Div+CSS+图片来创建漂亮的UI。

6. 更好的游戏状态控制。游戏有多种状态，如加载中、暂定中、游戏中……需要编写很好的管理器才能较好地控制和处理这些状态。

7. 性能优化。本文示例为了尽量简单些，几乎没有做任何优化。游戏优化是一个很大的话题，在这里就不展开讨论了。

以上只列出了一部分需要做的工作，要想开发一款真正的游戏，要做的还有很多。

HTML5游戏开发的现在和未来

得益于浏览器的高速发展，最近HTML5游戏开发的话题很火爆，以及移动平台对HTML5的广泛支持。

不过现在HTML5游戏开发依然面临着一些问题，例如：

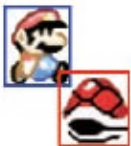
1. 可以很好支持HTML5技术的浏览器普及度还不够；
2. 各大现代浏览器对Canvas支持较好，但对Audio支持依然不力；
3. 在对标准的支持上，各个浏览器依然不统一，还需要进行跨浏览器的开发；
4. 性能依然不如Flash，而对Canvas的GPU加速实现的还不够广泛和完美；
5. 代码保护和防止作弊是难题。

除了以上几点，还有人反映开发HTML5游戏需要的技术太多，开发人员要掌握JavaScript、HTML5、CSS3。也许这是个问题，但对于广大Web前端开发人员来说，这些技术都不是难题。所以未来一段时间内，广大的Web开发人员依然是HTML5游戏开发的主力。

虽然HTML5游戏开发现在有很多问题，传统桌面电脑上竞争力也远不如Flash游戏，但是随着时间的推移，情况会有所改观。

而且随着更多支持HTML5，且搭载高频CPU/GPU、多核CPU移动设备的出现，HTML5游戏在未来移动领域一定会赢得更大的市场。

如果你和我一样，从开始编程的那天起，就有着开发游戏的梦想，而且恰巧你也有些Web开发的基础，那么不妨从此刻开始，让HTML5帮助我们梦想成真吧。P



注：本文所使用之图片资源，均来源于网络，版权归原作者。因篇幅所限，本文完整代码请参看《程序员》官网。

HTML5未来展望

■ 文 / Peter Lubbers / Ric Smith / Frank Salim / Brian Albers

我们将讨论一些还不够成熟但却极具潜力的HTML5特性。

HTML5的浏览器支持情况

如今，许多开发人员还在拼命开发所谓“健壮”的Web应用，耗费大量精力以兼容旧浏览器。截止2010年，在因特网中仍然存在着大量过时的浏览器，IE6就是这种残酷现状的代表。即便如此，IE6的寿命也是有限的，因为支持IE6的操作系统越来越难找了。总有一天，IE6的用户数会接近零。越来越多的IE用户已经升级到了新版本，而IE9将是Web发展的一个分水岭。无论何时，总会存在一个相对最旧的浏览器，而随着时间的推移，最旧的浏览器终会被淘汰。很多用户直接升级到了最先进的浏览器。未来的主流浏览器最起码应该支持HTML5的Video、Canvas和WebSocket，以及所有原来需要hack才能实现跨浏览器兼容的功能。

HTML5功能都已经基本稳定且被多数浏览器支持。除此之外，还有一些其他的HTML扩展功能及API尚处在开发初期。这里我们就来看一些尚未发布的功能。这其中有些处于前期试验阶段，而有些只需要进行细微修改即可形成最终规范并广泛推广。

HTML未来的发展

我们将讨论的几项激动人心的功能很可能在不久的将来就会出现在浏览器中。这些功能的实现可能也无需等到2022年。将来很可能没有HTML6规范，因为WHATWG曾暗示未来的规范将被统一简称为“HTML”。HTML将以增量的形式发展而非一蹴而就，其中特定的功能及其对应规范会单独发展。大家就这些功能达成共识之后，浏览器才会采纳并加以实现。在HTML5稳

定之前，一些即将发布的功能很可能已经在浏览器中大规模使用了。负责推动Web发展的组织为了满足用户和开发人员的需求，一直致力于Web平台的开发升级。

WebGL

WebGL是针对Web上3D图像的API。历史上，Mozilla、Opera、Google等浏览器厂商曾分别提供了试验性的JavaScript 3D API。今天，WebGL正向规范化方向迈进，而越来越多的HTML5浏览器加入了支持它的阵营。WebGL的规范化进程由浏览器厂商和The Khronos Group（负责OpenGL的组织，OpenGL是1992年创建的跨平台3D绘图标准）共同推进。OpenGL规范的当前版本为第4版，它已被广泛应用于游戏和计算机辅助设计中，并成为了微软Direct3D的有力竞争对手。

在Canvas元素中调用getContext(“2d”)，可以获得2D绘图的上下文。毫无疑问，这为其他类型的绘图上下文提供了方便之门。WebGL使用的同样是Canvas元素，只不过获取的是3D上下文。由于还在试验期，所以调用getContext()时，传入的参数需要使用浏览器厂商指定的名称作为前缀（moz-webgl、webkit-3d等）。例如，在支持WebGL的Firefox版本中，可以在canvas元素中调用getContext(“moz-webgl”)来获取3D上下文。这里调用getContext()返回的API对象与2D Canvas的有所不同，因为它提供的是OpenGL绑定，而非绘图操作。WebGL版本的Canvas上下文管理的是纹理和顶点缓冲区，而不是调用函数来绘制线条和填充形状。

3D HTML

与其他HTML5元素一样，WebGL将会成为

Web平台不可或缺的一部分。因为WebGL通过Canvas元素来渲染,所以它属于document对象。你可以像操作图像或2D canvas那样,在页面3D Canvas元素中应用定位和变换。实际上,任何在2D Canvas上能做的事情,在3D Canvas上都能做,比如叠放文本和视频、执行动画等。与纯粹的3D显示技术相比,结合其他文档元素,3D Canvas不仅可以用于创建HUD (Heads-Up Display, 平视显示器),而且使得2D与3D的混合界面变得更易于开发了。想象一下,在3D场景中利用HTML标签构造一个Web用户界面会是怎样的效果。许多OpenGL应用程序使用了非原生的菜单和控件,而WebGL则可以使用原生的样式美观的HTML5表单元素。

现有的Web网络架构也为WebGL的开发提供了便利。WebGL应用程序可以通过URL加载纹理、模型等资源。多人游戏可以基于WebSocket进行通信。例如,Google最近使用HTML5 WebSocket、Audio、WebGL等技术将经典的3D游戏Quake II移植到了Web上,并加入了多人竞争机制,如图1所示。游戏逻辑和图形使用JavaScript实现,页面呈现通过调用WebGL Canvas完成。游戏使用持久化的WebSocket连接来保持与服务器间的通信,从而实现对不同玩家位置的调整。

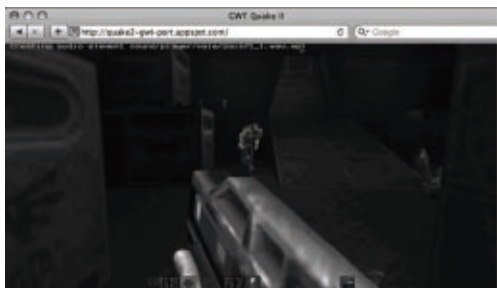


图1 Quake II

3D着色器

WebGL是OpenGL ES 2与JavaScript的结合,因此,它可以使用OpenGL中标准化的编程图形管道,包括着色器(shader)。着色器可将高度灵活的渲染效果应用于3D场景,让显示效果更真实。WebGL着色器是用GLSL (GL Shading Language, GL着色语言)编写的,这是Web中又一种新的专用语言。HTML5的WebGL应用程序使用HTML搭建框架,用CSS控制样式,用JavaScript处理逻辑,用GLSL进行着色。开发人

员可以借鉴在OpenGL着色器方面的开发经验,按照类似的方式使用WebGL API。

WebGL可以当做是Web 3D图形的基础层。与对DOM进行抽象并提供若干强大功能的JavaScript库类似,有一些库以WebGL为基础,提供了若干额外功能。正在开发的库涵盖了多个方面,有场景图,有3D文件格式(如COLLADA),还有游戏开发的完整引擎。图2显示的是Shader Toy,它是由Inigo Quilez搭建的WebGL着色工作台,其上提供的场景渲染效果是他与其他9名场景渲染艺术家一起设计的。这张截图显示的是Rgba的Leizex。我们希望在不久的将来,随着高级渲染库的蓬勃发展,即便是Web编程新手也能轻松地创建出3D场景来。



图2 Shader Toy是一个WebGL着色工作台

音频数据API

可编程的音频API与<audio>的关系类似于<canvas>与间的关系。在canvas标签出现之前,Web图像对脚本而言基本是不透明的。图像的创建和操作只能发生在服务器端。如今,已经有工具可以基于canvas元素来创建和操作可视化媒体了。同样地,在HTML5应用中可以使用音频数据API创建音乐。这将有助于丰富Web应用程序的内容创建能力,推进媒体创作工具与消费平台一体化(self-hosting)时代的到来。可以想象,将来总有一天,用户仅仅通过浏览器就能完成音频的创建和编辑。

简单的音频播放功能可使用<audio>元素来实现。但是,应用程序对音频的操作、分析或联机创建都需要底层API。若无法访问音频数据,那么诸如文本到语音的转换、语音到语音的转换、合成器、音乐可视化等功能都是空谈。

将来,标准音频API不仅有望操作数据元

素中的麦克风输入，而且还有可能操作<audio>标签内引入的文件。通过<device>和音频数据API，可以构建一个允许用户在页面内录制和编辑声音的HTML5应用程序。音频片段可存储在本地浏览器缓存中以便重用，还可与基于canvas的编辑工具相结合。

对此，目前Mozilla的每夜构建版本中有一种试验性的实现方式。Mozilla音频数据API可被当做实现标准跨浏览器音频编程功能的基础。

视频元素改进

在HTML5仍然迟迟未能指定必选视频编解码器的时候，Google最近发布了一种面向Web的高质量、免费的网络视频编解码器——WebM。虽然并不是所有厂商都支持WebM，但这种格式背后的支持力量还是很强大的。

此外，对当前HTML5视频控件进行改进的需求也与日俱增。为了全面取代基于插件的视频展现方式，HTML自身控件需要引入更好的用于流媒体的API，期待这种能力能在下一代媒体标签中有所发挥。不支持Flash的设备越来越多，如Apple iPad和iPhone，所以要让大众用户正常浏览网络视频，只能依靠HTML5，这是HTML5发展的重要契机。

触摸屏设备事件

Web访问方式越来越多地从台式机和笔记本电脑转换到了手机和平板电脑上，因此HTML5的交互处理方式也在逐渐发生变化。Apple在推出iPhone的同时，也将一系列特殊事件引入到了其浏览器当中，这些事件用来处理多点触摸输入和设备旋转。尽管还未被标准化，但这些事件已被其他移动设备厂商选用。学会此类事件，就可以针对当今最先进的设备开发出合适的Web应用程序。

方向事件

方向事件是移动设备中最简单的事件，它可以加入到页面body标签中：

```
<body onorientationchange=" rotateDisplay();" >
```

在方向事件处理程序中，可以引用window.orientation属性。该属性可选的值如表1所示，它们以页面首次加载时设备的方向为基准。

表1 方向值及其含义

方向值	含义
0	页面当前方向与首次加载时的原始方向一样
-90	与原始方向相比，设备顺时针旋转了90度（向右）
180	与原始方向相比，设备旋转了180度（垂直翻转）
90	与原始方向相比，设备逆时针旋转了90度（向左）

手势事件

移动设备支持的另一种事件相对高级一些，称为手势事件。手势事件可以理解为通过多点触摸引发的缩放或旋转。当用户有两个或多个手指同时在触摸屏上挤压（pinch）或扭转（twist）时，就会触发手势事件。扭转表示旋转，挤压（pinch in）和伸展（pinch out）分别表示缩小和放大。为了接收到手势事件，代码需要注册如表2中所示的事件处理程序。

表2 手势事件处理程序

事件处理程序	描述
ongesturestart	用户将多个手指放在触摸屏上，并开始滑动
ongesturechange	用户正在使用手指动作进行缩放或是旋转操作
ongestureend	用户移开手指，缩放或旋转操作已经完成

在用户做手势的过程中，事件处理程序会灵活检测事件的缩放或旋转属性，并对显示效果进行相应更新。以下代码是手势处理函数的示例。

```
function gestureChange(event) {
    // 获取用户手势生成的缩放量
    // 1.0代表初始大小，小于1.0表示缩小，大于1.0表示放大
    // 基于尺寸大小按比例放大和缩小
    var scale = event.scale;

    // 获取用户手势生成的旋转量
    // 旋转值介于0度到360度之间
    // 正值代表顺时针旋转，负值代表逆时针方向旋转
    var rotation = event.rotation;

    // 基于旋转操作更新页面显示
}
```

```
// 为文档节点添加手势变换监听程序
node.addEventListener("gesturechange",
    gestureChange, false);
```

在需要操作物件或者以显示为主的应用程序（如绘图工具和导航工具等）中，经常需要用到手势事件。

触摸事件

如果需要在低层次上处理设备事件，可以通过触摸事件获取所需信息。表3所示为不同的触

表3 触摸事件

事件处理程序	描述
ontouchstart	已经在触摸设备表面放置了一个手指。当多个手指放在设备上时，会发生多点触摸事件
ontouchmove	在拖动操作中，一个或多个手指发生了移动
ontouchend	一个或多个手指离开设备表面
ontouchcancel	意外中断停止了触摸操作

摸事件。

与其他移动设备事件不同，触摸事件需要考虑多点数据（多个手指可能会同时触摸）同时出现的情况。因此，用于处理触摸事件的API会相对复杂一些，如下代码所示。

```
function touchMove(event) {
    // touches变量是一个列表，包含了当前每个手指的触摸点信息
    var touches = event.touches;

    // changedTouches列表包含当前触摸状态发生变化的手指的触摸点信息，如添加、移开或重放手指
    var changedTouches = event.changedTouches;

    // 监听器注册在哪个节点上，targetTouches就包含哪个节点上发生的触摸操作的触摸点信息
    var targetTouches = event.targetTouches;

    // 在取得预定的触摸点之后，可以取得能够从其他事件对象中正常获取的大部分信息
    var firstTouch = touches[0];
    var firstTouchX = firstTouch.pageX;
    var firstTouchY = firstTouch.pageY;
}

// 为我们的示例注册一个触摸事件监听器
node.addEventListener("touchmove", touchMove, false);
```

不难发现，设备的原生事件处理机制可能会干扰手势事件和触摸事件的处理。在这种情况下，可调用：

```
event.preventDefault();
```

上述代码会覆盖浏览器界面的默认行为，让开发人员自行处理事件。在所有移动事件被标准化之前，建议开发人员先详细阅读相应设备的开发文档，再进行开发。

P2P网络

在Web应用程序中，高级网络技术始终在向前发展。不论是HTTP还是WebSocket，都有客户端（浏览器或其他用户代理）和服务器端（URL主机）。P2P网络允许客户端之间直接通信。通常情况下，这比从服务器发送数据更高效，而且有助于降低托管成本，提高应用的性能。要想开发更快的多人游戏和协作软件，P2P技术是不二之选。

在HTML5中，P2P结合device元素可以构建高效视频聊天应用。在P2P视频聊天应用中，对话双方能够直接互传数据，无需通过中心服务器。在HTML5之外，P2P视频聊天已广泛应用于诸如Skype等应用程序中。由于流媒体视频对带宽要求比较高，如果没有P2P通信技术的话，很可能那些流媒体应用程序一个都无法实现。

针对P2P网络，浏览器厂商也进行了一些试验，例如Opera的Unite技术，该技术会直接在浏览器中部署一个简易的Web服务器。Opera Unite允许用户为别人创建和提供服务，从而实现聊天、文件共享和文档协作等功能。

当然，面向Web的P2P网络不仅需要考虑如何为开发人员提供编程API，还需要引入一个兼顾安全和网络媒介的协议。

最终方向

到目前为止，我们一直着眼于如何让开发人员构建强大的HTML5应用程序。我们考虑到的另一种视角是基于HTML5的Web应用如何能让用户获得上佳的体验。许多HTML5功能旨在消除或降低脚本的复杂性，提供之前需要插件才能实现的功能。拿HTML5的video元素来说，无需任何JavaScript即可指定控制界面、自动播放、缓冲行为和生成并显示缩略图。而通过CSS3可以直接在样式文件中实现以前用脚本才能实现的动画效果。这些声明式代码让应用程序更容易被用户理解，并最终让用户获得更多掌控。

你可能已经见过Firefox和WebKit的开发工具是怎样揭示HTML5各种功能信息的，包括存储功能，以及至关重要的JavaScript调试、性能分析、命令行求值等功能。因此，简单的声明性代码、浏览器内置或Web应用程序自带的轻量级工具将是HTML5未来的发展方向。

小结

各位读者可能接触Web或者从事Web开发超过十年的时间了，回想一下最近几年HTML技术带来了多大的变化。十年前，“专业HTML编程”意味着学习HTML4的新功能。当时顶尖的开发者刚刚发现动态页面更新和XMLHttpRequest。虽然Ajax技术得到了广泛关注，但那也是在发明“Ajax”这个术语几年以后的事了。当时，浏览器中的大多数专业编程还只是限于讨论框架和操作图像地图而已。

曾经需要数页脚本代码才能实现的功能，如今只需一个标签就能轻松搞定。只要愿意，你随时都可以下载一个支持HTML5的浏览器，打开自己喜欢的文本编辑器，亲自动手尝试一下令人心驰神往的HTML5编程吧！



本文摘自《HTML5高级程序设计》（Pro HTML 5 Programming）一书，中文版由人民邮电出版社图灵公司推出，特此感谢图灵公司的授权支持。

寻找机遇 创造未来 —— CSDN 热门职位全新推荐

方正国际软件有限公司

诚聘英才 FOUNDER 方正

北京：医疗行业咨询顾问
产品经理（医院信息化）
CIS、LISP、PACS/RIS、药品管理、手麻方案设计

苏州：.NET软件工程师
Java软件工程师
方正飞鸿产品售前顾问（北京、天津、大连、成都、西安、武汉、济南、深圳）

详细信息请登录CSDN网站查询！

公司主页：<http://www.founderinternational.com/index.html>
电话：0512-86665500-6506 邮箱：srhr@founder.com
联系人：牛小姐

Zynga Beijing



长期招聘职位：

PHP软件开发工程师
Flash软件开发工程师
数据分析工程师
网站系统管理员

更多职位招聘信息和详细职位要求可登陆智联招聘或中华英才网查询！

公司主页：www.zynga.com
地址：朝阳区远洋光华国际C座10层
加上：请以“职位+姓名+CSDN”形式命名的邮件，将简历发送至：BeijingJobs@zynga.com

欧特克软件（中国）有限公司

Autodesk® 你一展身手的舞台，打造美好未来

你一展身手的舞台，打造美好未来加入我们的队伍，以一种全新的方式实现设计理念。无论是设计可持续性建筑还是开发混合动力汽车，Autodesk软件都可助你一臂之力。作为全球领先的二维与三维设计、工程和娱乐软件提供商，我们致力于帮助打造出更美好的未来。

让我们共同创造未来！
更多职位信息请浏览以下网站：www.autodesk.com/careers
或直接将简历发送至：acrd.hiring@autodesk.com

广州瀚信通信科技股份有限公司



岗位：Windows C#/C++开发工程师

工作地点：广州

职位描述：专注于移动通信信令类软件和系统平台产品开发
任职要求：

1. 2年以上使用C#/C++开发Windows应用程序的经验；2. 熟悉Windows应用程序开发，熟悉C#.NET或C++，熟悉Visual Studio开发工具；3. 了解ADO.NET数据库访问技术；4. 熟悉SQL语言；5. 具备英文阅读能力，学习能力强，思维敏捷；6. 为人正直，工作勤奋主动，有责任心；7. 具有良好的沟通能力和团队精神。

请以“职位+姓名+CSDN”作为应聘邮件标题，将简历发送至：yfhr.list@hantele.com
公司主页：www.hantele.com

搜狐畅游



游戏研发部3D程序专员

工作职责：

1. 3D框架下各种功能模块和算法的实现；
2. 研究、开发和优化3D引擎。

职位要求：

1. 诚信正直，具备高度的责任心；
2. 计算机相关专业本科及以上学历；
3. 两年以上的3D引擎开发经验；
4. 精通C++，面向对象编程，Windows编程；
5. 熟悉3D图形学原理，熟悉3D几何；
6. 精通DirectX/OpenGL，精通DX9；
7. 具备良好的算法和数学基础，优秀的编码习惯和学习能力。

公司网址：<http://www.changyou.com>
E-mail：gamejob@job.cyou-inc.com

世纪乐知（北京）网络技术有限公司

杂志广告销售



岗位职责：

1. 负责《程序员》杂志广告销售，CTO专刊广告销售；
2. 与商务部其他同事一起协作承担商务部业绩指标；
3. 负责新客户开拓，及客户关系维护。

任职要求：

1. 从事过广告销售3年以上，平面媒体广告工作者优先；
2. 良好的商务意识，良好的职业道德及职业素养，积极上进；
3. 沟通能力强，热爱销售工作，敢于接受挑战，积极上进；
4. 有IT行业从业背景者优先考虑，英文良好者优先考虑。

电子邮箱：hrzhaopin@csdn.net

我的编程竞赛之路

——中国大学生计算机编程第一人楼天城访谈

记者 / 陈秋歌

25岁的楼天城有“中国大学生计算机编程第一人”的称号，也被参加竞赛的学子们敬称为“楼教主”。他的传奇经历一直激励着众多年轻学子：从2001年开始参加计算机编程竞赛，并连获全国一等奖；2004年入选国家集训队，同年获得IOI金奖并被保送到清华大学；2008年开始在首位亚裔图灵奖获得者姚期智院士的理论计算机中心攻读博士；2007年和2009年两次带队参加ACM/ICPC全球总决赛并获得第二名。《程序员》近日专访了编程奇才楼天城，让他与我们分享一下他传奇的编程竞赛之路。

记者：请简单谈谈你的童年经历。童年哪些人、哪些事对你的影响最大？

楼天城：小时候，我比较爱玩，并不怎么

喜欢学习。后来在8岁那年，我跟随父母一起到清华大学游玩。身处中国的最高学府，我不自觉地受到了感染，当时就给自己定下了童年的第一个目标——要在学习上投入更多的精力，严格要求自己，将来一定要到清华大学读书。这是童年中对我影响最大的一件事儿。经过十年的努力，现在这个愿望已经实现了。

父母是童年时代对我影响最大的人，他们特别注重对我心理、性格等素质的培养，而不是局限于学业。他们认为高素质对人的影响远远超过学习能力的影响。在高中我能如愿参加各项编程竞赛，也离不开父母的支持，但前提是数学等智力科目不会受到影响。这类竞赛并不属于高考科目，而且势必会对学业产生一定影响，如果没有父母的支持，参加竞赛这条路是走不通的。父母的支持支撑着我一直在竞赛这条路上走下去。

另外，我比较喜欢各种棋类，还喜欢踢足球。棋类可以锻炼人的思维，而运动也会培养人的思考能力。小时候我就考虑，向对方球门底线带球（如果是向球门带球，一定是越向前越好），球门与球的角度是越变越大还是越变越小；往前带球，这个角度是越大越好，还是越小越好。经过计算后，我发现把球带到某个地方时，角度是最大的，再往前带就会变小，而这个地方就是射门的最佳位置。结论是：这个最大的角度与距球门的距离有关。

记者：你是什么时候开始接触编程的？在学习编程过程中，都开发过哪些应用？

楼天城：我从高一开始学习编程相关知



楼天城在2010 topcoder公开赛现场

识。编程对于高中生来说，确实是一门特别的科目。我所开发的应用中印象比较深刻的是黑白棋。当时在联众的游戏平台上，它有90%的胜率。最初设计的黑白棋只能搜索到六层，它的“棋艺”并不理想，当时很多此类的黑白棋游戏也能做到这一步。为了突破这一局限，我研究了各种优化技术、博弈算法及搜索技术，对算法进行了优化，使原来的搜索增加了四层，它的计算能力、博弈水平也随之提高了很多，一时很多人都难以胜过它。虽然这个游戏并没有像李开复当年写的黑白棋那么优秀，但对我来说，还是蛮有成就感的。另外，我还做过很多其他的应用，但大多是实现一些涉及核心的关键问题，并没有作为产品公布出来。

记者：你第一次参加编程竞赛是什么时候？在之后的竞赛经历中，遇到了哪些挑战？

楼天城：2006年，我第一次参加编程竞赛，是Google编程挑战赛。由于一个失误，我与冠军擦肩而过。大赛结束后，我非常沮丧。后来在父母和老师的鼓励下，我下定决心要再等机会，赢取冠军。

两年后，直到2008年，我才有机会再次参加Google编程挑战赛，并在最后8分钟赶超了当时领先于我2小时52分的南非选手，获得了冠军。这次大赛对我来说可以算作一次真正的挑战。

记者：2010年举办的topcoder公开赛，你取得了一个不错的成绩，对此有什么感受？

楼天城：topcoder公开赛是世界范围内一年一度的编程大赛，2010年在赌城拉斯维加斯举行。我参加的其中两项不同的比赛都进入了决赛，一项是24小时的类似马拉松的Coding比赛，另一项是算法编程。这次比赛比较可惜的是，在最后决赛阶段犯了一个小小的错误，而滑到了亚军的位置。以后再有同样的机会，争取能够获得更好的成绩。

记者：请谈一下现在读博的感受。

楼天城：我从2008年开始在姚期智院士领导的清华大学理论计算机中心攻读博士学位，现在还处于学习和提高阶段，研究方向主要是以组合算法为主。总之，很幸运能够得到姚院士的指导，这段时光注定将成为我人生中最重要阶段之一，我会用心努力做好每一天。

记者：除了编程竞赛，你之前还参加过很多数学方面的竞赛。有些人认为编程与数学没



团队协作，志在ACM大赛桂冠

有太大的关系，你怎么看？

楼天城：可能是我们所遇到的问题不太一样吧。我遇到的问题多侧重于核心部分，往往比较抽象，对效率要求也很高，要解决它们，必须有深厚的数学基础，从这个方面来说对数学要求还是非常高的。

记者：你怎样看待编程竞赛？对有志于投身编程竞赛的人有何建议？

楼天城：编程竞赛是一个“公平、刺激、酷”的脑力游戏，除了能获得乐趣之外，还能很好地锻炼自己的思维，磨炼坚韧随和的性格。参加了多次竞赛，我真实地体会到它的残酷性。编程竞赛是顶尖高手之间全方位素质的较量，即使一时的状态不佳，也会让你与冠军失之交臂，最终换来的将是几年的漫长等待。但也相当刺激、公平。这也是我喜欢它的主要原因。

我感觉自己很幸运，能够接触到这些编程竞赛。这种比赛形式，完全符合我的思维方式。在这十多年来，无论在思维的培养还是对性格的磨炼上，它对我都起了很大的作用。

至于建议，像我们这些靠理性思维来解决问题的人，平时过于依赖自己的计算能力，不相信定式、经验之类的东西。而实际情况却是，有一些凭借经验才能做到的事情，我们做得不够好，只是想当然地以为计算等于一切，结果却与目标相差很远。所以有很多非智力因素也需要去掌握。否则，你可能会因此而落后于他人。这算是我的一个建议。

另一个建议就是不要把参加竞赛当作上大学的一个途径。竞赛可以帮助你提高自学能力和综合素质，锻炼思维能力，也可以帮助你提高其他科目，但平时一定要注意协调好竞赛与其他科目之间的关系，不要把其他科目落下，不要让竞赛成为影响正业的事情。📌

Facebook工程管理揭秘之二

让亲身实践者执行工作流程

文 / 黄易山

Facebook前工程总监黄易山（Yishan Wong）撰写了一系列文章，很好地总结了Facebook卓越研发文化中的宝贵经验。本刊将陆续连载这一系列，本文是第二篇。



黄易山

1997年毕业于卡内基-梅隆大学。2001年加入PayPal，曾任高级工程总监。2005-2010年在Facebook领导研发，在公司研发环境的建设上发挥了重要作用。

在这里，我使用“工作流程”这个词来描述“个人或团体为了完成一项活动而遵循的步骤”意义上的流程，以及组织的一般制度。

随着一家公司的成长，有必要增加或整理工作流程。最重要的利弊权衡通常是工作流程所带来的阻力，以及效率或效益上的收益孰轻孰重。

一方面，很难评估这种权衡中的利弊，因为其中牵涉到很多因素，所以有一条可能会有帮助的原则：只允许那些有特殊需要的工作流程被执行，而且要由那些直接使用它的人来执行。通常，经理和管理人员会提议工作流程，因为它会帮助他们更好地指挥、控制、协调或沟通。但新工作流程的执行不应该为这些目标服务，因为它的收益是不实际的，而且往往被高估：管理人员可以看到它带来的好处，但由于至少在一个（或多个）层面上不需要他们的直接操作，所以他们没有认识到它的真实成本。

另一方面，那些亲自动手的人（如工程师）可以很容易地辨别，何时需要适当补充组织要素或工作流程，因为他们可以更直接地认识到收益将大于成本。只有在这种时候，才应该在组织中增加一个新制度或工作流程。

经理们必须抑制住自己天生的恐惧，不要害怕由于缺乏对细节的能见度而造成的混乱或失控。他们应该重视依靠建立准确、翔实的背景情况和目标来进行领导，促进技术工人（不知为何，他们通常不是那种容易陷入无政府状

态的人）之间互相合作而形成的自然组织。

影响

个人制定并执行的工作流程，会针对真实工作的情况进行更多优化调整。管理者设计的工作流程，最多能接近实际工作流程，它需要管理、优化或整理。这是许多愚蠢、低效的工作流程的来源。

在人们亲自制定工作流程时，会感到更大的自主权。在今后，随着情况不可避免地变化，也就更有权视情况对其调整，而不是任其僵化。外部强加的（自上而下的）工作流程更加难以打破，而且往往会被神化，从而产生非常大的组织惯性。

对协调和沟通的影响

对于协调和沟通，会有什么影响？这些不是管理者的职责所在吗？是的，这是他们的重要职能之一。然而，为此而创建的工作流程应该由经理来执行，而不能强加在普通员工身上。

例如，团队状况和项目更新的广播是一种有用的功能，可以通知该部门或公司的其他人哪些事情正在发生。这是一个完全可以由管理人员完成的工作流程，并不需要团队中其他人的直接参与：经理可以撰写这些更新并分发给团队成员，但他不应该反过来要求团队成员也做同样的事情，因为经理与他们一同工作，可以（也应该）知晓团队运行的状况。

这里的关键是，我们有一个有益于管理人员的工作流程，而这其中的成本也由他们承担，从而允许他们为自己的共同利益对工作流程进行优化。这不需要（也不应该）涉及到其他任何人。

描述性工作流程的微妙危险

指令性的工作流程意思是“这是为了完成X，你必须采取的步骤”，描述性的工作流程则是“今天我们完成X后，让我们记录下采取过的步骤”。首先，描述性的工作流程是一个普通的、看似无害的建议，但不可避免地会导致同样的不必要的工作流程要求和僵化的组织制度，因此我们应该同样积极地避免使用它。

首先，写下“我们通过这些步骤，完成了X”（其中的X可能是“推出一款产品”或“将一个想法从概念变为现实”或“申请一张新桌子”）的无害建议被采纳，而且这个工作流程被归档成一份文件，然后（也许）会被张贴在公司内部网络上。

然后有一天，一名新人问道：“我们该怎么完成X？”在回答时，我们不再像老前辈那样非正式地解释完成X要如何如何，而是让他们去参考那份文件。毕竟，这比口头解释更简单快捷。在快速成长的组织中，新人加入的速度很快；很快地，这名新人变成了老前辈。另一名新人碰巧问他们同样的问题，这次轮到他们让别人去参考那份文件了。这个过程重复循环几次，该文件就被视为完成X的权威指南了。

在公司现有的大多数人（在快速成长的公司中，新人总是比老前辈多）的概念中，原创的具有适应性的工作流程（适应性强、有机和运作良好，而且它的非正式使人们觉得它灵活）被文件所取代，文件现在被看成做某事的非常精确的规范。

巧妙、可控制的灵活性减少，而且描述性工作流程成为了规范，但这无法归咎任何人。有时会更糟，因为有意的指令性工作流程存在的背后通常有一个权威的、能够亲自站出来呼吁的特定的人支持。僵化的原创性工作流程存在的背后有一篇单独的文件支持，给人们一种无法形容的类似法律的权威感，也就是说，超越任何一个个人的权威，在必要时就更加难以推翻和创新。

长期影响

在不同时期，我都曾参与过对潜在收购目标的慎重调查。最令人惊讶的发现之一是，在许多情况下，一个相对较小的公司却有更多的正规工作流程，这主要是由它们自己的员工所导致的，也是他们发展相对缓慢的原因：很少想法能得到执行，并最终导致了相对弱势的市场地位（有时，这是我们认真考虑某次收购的原因）。

在Facebook，公司文化与工作流程是有抵触的，引入新工作流程的通常模式是“只有在事情快要不可收拾的时候，才会考虑引入新的工作流程”。尽可能地做到这一点，而且可能要更变本加厉，因为这样你所得到的回报将是整个公司的高效率。如果你的公司比其他同等规模的公司拥有更少的工作流程，你的创新和执行速度会更

在Facebook，公司文化与工作流程是有抵触的，引入新工作流程的通常模式是“只有在事情快要不可收拾的时候，才会考虑引入新的工作流程”。

快，将想法从一个概念到最后推向市场也会更加快速。在内心深处，经理可能要和更多让他们不安心的混乱作斗争，但是让人不安心的混乱和真正对公司造成威胁的混乱有着很大的区别。越接近后一时刻，就越能够对比出技术行业的最大的优势之一：执行速度。

工作流程通常建立在有规律的和大致固定的执行速度的基础上。因此，形成这种执行速度是保持长期高效的关键。如果你的公司有特定数量的工作流程，而且比其他同等规模的公司要少，那么你的执行速度会更快，而当你的公司变得异常庞大时，其中的官僚主义也比较少，这同样会产生倍增效果：当公司规模较小时，执行速度快两倍，意味着同一件事，你需要两个星期完成，而你的竞争对手需要四个星期；而一旦公司规模扩大，你可以在两年内完成的一件事，你的竞争对手需要另外两年才能迎头赶上。这多出来的两年对他们来说，可能意味着末日。🔴

模型驱动开发

——在RUP与Agile之间找到平衡点

文 / 姚冬

本文介绍了集统一软件开发和敏捷开发方法优点于一体的新型软件开发方法——模型驱动开发。

当今是一个快速发展的时代，软件的功能更强大，应用更广泛，系统架构更复杂。与此同时，软件开发的难度也越来越大，软件质量难以得到保障。在与业界同行交流的过程中，我感受到更多的不是自信，而是对软件质量的无可奈何与力不从心。

为了解决软件开发存在的问题，业界不断涌现出许多开发方法、过程以及模型，试图从方法论、工程学等角度对软件开发过程进行改进和管理。其中最为知名的，要数RUP（Rational Unified Process，统一软件开发过程）和近十年涌现的各种Agile（敏捷）开发方法了。但从实际效果来看，似乎都没有达到预期的目的。

本文试图以模型驱动开发（MDD）为契合点，在RUP的坚实与Agile的灵动之间找到一种平衡。

模型驱动开发概述

模型驱动开发就是以“模型”为核心的软件开发过程，业界存在许多特定领域的建模语言（如DoDAF、AUTOSAR等）。但是以UML最为通用、规范和科学化，成为业界建模事实上的标准。

模型驱动开发专注于业务的建模和抽象，而不是具体的语言和算法。这种软件开发方法使开发人员把更多精力放在产品的需求分析和功能划分上，而不是纠缠于软件的实现细节。

模型以图形化的方式来表述所研究的系统、过程、事物或概念，而图形化的方式更符合人脑的思维方式。

用文字去表述的概念，例如软件需求，很容易造成误解。相比较人类的语言文字，模型在表述时更加准确，尤其是标准化的建模语言，例如UML。

模型驱动的开发方法使软件实现和模型设计完美地统一起来。软件发展的模型驱动方法不是一种全新的方法，随着UML2.0（目前最新版本是2.3）及相关工具的推出，其支持性技术越来越成熟，也愈加受到重视。

模型驱动开发的成熟，不仅体现在效率方面，更体现在可测量性方面。这种能力是与方法及工具密不可分的。UML2.0版本在语言定义精确程度有了相当的提高，是模型驱动开发的必要条件。UML2.0从通信界代码生成技术最成功的SDL语言汲取了大量语法规则，使得代码完全自动生成成为可能。自动化意味着模型转换时的不明确和不精密的消除，同时能够保持良好的一致性，使由模型直接生成最终代码成为切实可行的方案，也为最终生成代码的质量提供了保障。同时对特定领域的改进和支持也

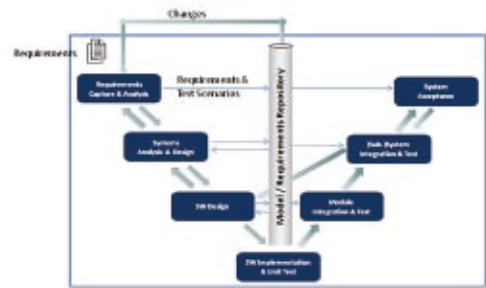


图1 模型驱动开发过程

使得UML能够更好地适应各种应用需要，例如嵌入式领域。

从图1可以看出，软件开发的整个过程都是围绕着模型与需求展开的，这里有三个驱动力。

需求驱动的开发：以需求为核心，需求是关键，是第一驱动力。一切任务都来源于需求，需求是定义开发范围，验证开发结果的唯一标准。

模型驱动的开发：建模活动贯穿始终，模型是我们工作的方式，通过模型去描述需求和设计系统，进行具体的实现，甚至通过模型进行测试以验证我们的设计和实现。

测试驱动的开发：测试驱动用来验证各阶段任务是否正确合理，保障各阶段始终保持一致，测试过程并不是只存在于开发的后期，而是分布于开发过程的每一个阶段。

可以说，需求、模型、测试，三者相互关联，贯穿开发生命周期的各个阶段，有机地推动开发过程向前发展。

模型驱动的开发过程，也是迭代的开发过程。在需求分析、系统设计阶段，我们可以随时进行验证和迭代。在软件具体的开发阶段，我们按不同的粒度，进行多次迭代，使一个周期就是从设计到开发再到测试的过程。

模型驱动开发与RUP、Agile

在模型驱动开发的实践中，已经总结出了一些需要遵循的基本原则，很多原则都与RUP以及Agile的思路一致，从中我们也可以看到，模型驱动开发与RUP和Agile的关联。

- 可工作的软件为核心和评估标准。
- 模型驱动。
- 计划，追踪与调整。
- 持续反馈，重视质量及优化。
- 持续的代码生成。
- 持续的调试与测试。
- 持续集成。
- 测试驱动开发。

模型的天然特性结合相关工具的支持，既可以借助RUP完整定义的流程，又可以有效地实施敏捷开发的最佳实践。通过模型建模，快速地构建系统框架，建立起快速原型，并借

助模型的仿真和执行，来模拟目标系统的运行行为。这种方式可以尽早将软件系统原型与客户进行沟通 and 呈现，更形象化地与客户进行交流，快速捕获客户的需求和反馈，进而指导下一步开发和下一个原型的产生。

这使得“以更低的成本交付高质量系统”成为事实上的可能。

模型驱动与RUP的关联

UML是模型驱动开发中最常见的语言，它本身就与RUP一脉相传，因此模型驱动开发与RUP关联紧密。RUP强调的软件开发过程，在MDD模型驱动开发中同样适用，而且会遵循得更加有效。

● 迭代的开发

模型的分层构架，从不同层面、不同角度对系统进行描述，通过UML对系统良好的描述和内建的关联，使迭代开发成为必然。借助工具的仿真和执行，我们可以在不同阶段对模型设计加以验证，而每一次的迭代，都可以由模型直接生成可运行的原型。

图2是一个从分析到开发再到测试的以模型为中心的连贯的迭代过程，它关注持续集成，同时测试驱动的理念贯穿始终。

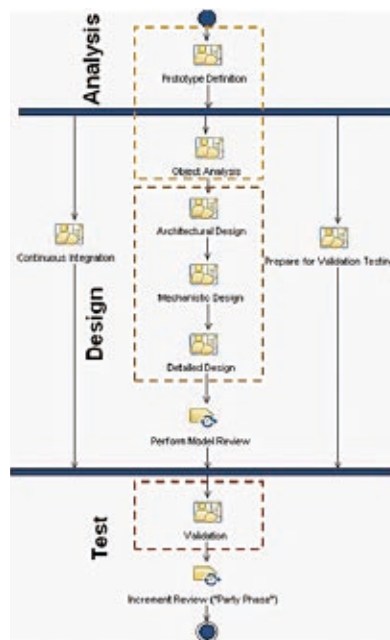


图2 模型驱动的迭代过程

文章点评：

首先，我并不认为模型驱动开发已经成为流行的开发模式，这一构想在方法学上仍有很多难于实施的地方。其次，我并不认为建模语言比编程语言表述更准确，编程语言有其严密而精确的正确性论证，但即使如此，一个再高明的开发者也会写出不可理喻的错误代码，这对建模语言所谓的“精确性”是同样的挑战。最后，仅从本文的探索来说，我认为作者的某些不严谨的陈述仍显出矛盾的核心，即：项目全程建模的结果与工程实施的目标分别是模型与产品，二者仍然有相当远的距离。这即是说，即使模型与代码之间的转换成本为零（当然，目前这仍然是梦想），那么转换的结果仍然不是产品。进而说，MDD并不是产品开发的解决之道。以上，尝作一家之言。

然而作者在本文中的尝试，仍然可以为工程界开辟思路，即：在新的理论面前，旧的方法究竟是主动去融合他们，还是守住理论上的严密、谨慎求变呢？这是我对阅读本文的一点建议。

点评人：资深软件工程师、技术作家周爱民

●需求管理

需求同样可以进行建模，并且能够在工具中管理起来，进行需求的关联分析、覆盖度分析、变更影响评估、需求的回溯、需求风险分析，从而使得需求管理变得容易，实现从软件开发的根源把控质量。

●基于构件的体系结构

工具支持由模型生成不同架构的目标代码，模型又可以支持不同的设计模式，甚至有专门针对模型开发的设计模式，有效地保障了生成代码的质量，并且良好地支持模型的可重用性。

●可视化软件建模

在这方面，RUP与模型驱动开发是一致的。不同的是，MDD工具的出现，使得可视化建模变得更简单、更合理，而且可以支持整个生命周期的开发建模，从需求捕获，到系统设计开发、模型测试、代码生成，都可以围绕建模展开。

●验证软件质量

软件质量通常由测试来保障，而传统软件开发之所以测试不利，一个原因是不能更早地介入，无法在早期阶段针对需求分析和设计进行测试验证。而借助工具的支持，模型驱动开发得以将模型仿真，支持设计的验证，生成快速原型，及时有效地与客户交流。

●控制软件变更

软件变更的控制，一方面需要评估变更造成的影响，这一点通过模型之间建立起来的关联可以轻易达到；另一方面需要测试验证变更涉及模块，保证修改在准确实现变更的同时，不影响其他功能模块。这可以通过工具支持的模型回归测试等手段实现。

MDD对敏捷开发的借鉴

模型驱动开发在很多方面与敏捷开发的思想非常契合，而模型自身的特点，也使得它在实践敏捷开发多种实践时来得更容易、更自然、效果更好。

●持续迭代，持续反馈

使用短周期的迭代可以加速开发过程，例如Scrum的一次Sprint。基于开发出的快速原型，客户可以更好地理解自己的需求，开发者

也能了解到如何才能更好地满足客户的需求。模型的可执行性决定了模型驱动的开发可以快速地生成可执行的原型，从而使得持续迭代成为简单的事情，与客户的持续反馈也变得更加容易沟通和理解。

通常一次迭代会由一系列任务组成，而每一个任务的进展都可以看作是一次更短周期的迭代过程，如图3所示。需要强调的是，其中的分析设计与实现以及测试和执行都是围绕模型展开的。

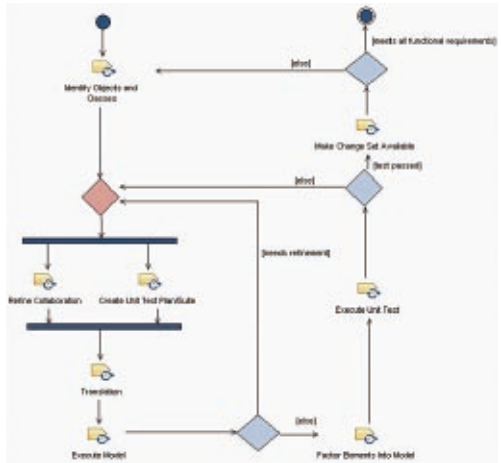


图3 模型驱动开发中的任务实现过程

●可工作的软件

模型可执行性决定了设计、开发出来的软件是天生可执行起来的，而是否正确工作则完全可以通过模型的仿真进行验证，模型驱动开发的原则之一就是可以工作的软件作为核心和评估标准。

●响应变化

我们从不同层面不同角度对系统进行建模，建立起模型之间的关联，在系统发生变更时，可以便捷地发现变更所造成的影响，同时通过模型的仿真与验证快速验证我们的改动，从而快速地响应变化。

●测试驱动

模型驱动的测试是部署在整个软件开发生命周期的，从系统需求的捕获阶段、系统设计到各个测试阶段，都可以通过执行模型来进行验证。

●持续集成、持续构建、持续测试

通过工具软件，我们可以很方便地进行持

续集成、持续构建，而模型既可作为开发建模的输出，又可作为测试用例的输入来源，测试的进行也同样可以依赖于工具自动进行。

● 团队协作

整个开发团队是以需求为核心、以模型为活动方式展开的。UML模型可以适应不同规模、不同复杂程度的系统，从而使得围绕在模型驱动周围的开发团队也同样可以适应不同的团队规模。所有开发活动都可以围绕着模型构架进行，模型构架的层次性使开发团队的分工和合作变得简单。

结束语

模型驱动开发是一种新的软件开发方法，它结合了RUP和Agile开发的优点，借助强有力的工具支持，可以良好地实施多种最佳实践。

需要强调的是，模型驱动开发不是万能良药，企业和开发团队在实施的过程中，应当根据自身情况以及项目需要，进行量体裁衣。而且由于是基于模型的开发，使我们在学习和适应模型上需要时间的投入，在实践的初期很可能效率上反而不如原有模式。这时就需要一点坚持和投入的决心。

完全有理由相信，模型驱动开发将是推动软件开发向前迈进的强有力支撑。也希望越来越多的行业人员能够了解、熟悉并对其加以实施推广，模型驱动开发的道路将越走越广！

参考文献

《统一软件开发过程》（美）Ivar Jacobson, Grady Booch, James Rumbaugh等著，周伯生，马学民，樊东平等译，机械工业出版社 ISBN 7-111-07572-2/TP.1200

“Agile Model-Driven Development for Real-Time and Embedded Systems with Rational Harmony” Bruce Powel Douglass



姚冬

北京邮电大学软件工程在职研究生。现就职于某知名跨国企业，关注复杂系统开发，以及软件过程改进等方面的研究。拥有12年的系统软件开发和软件工程实践经历。

责任编辑：董世晓（dongsx@csdn.net）

新闻 | News

基于“Loong”平台核高基中间件发布

3月4日，“核高基”国家科技重大专项“国产中间件参考实现及产品”成果发布会暨“四方国件”联盟工作启动会在北京召开。会上发布的“Loong”平台是自主可控、具有自主知识产权的国产中间件产品，可以支撑大型网络应用系统实现高效运行、快速开发和灵活部署。

IBM大中华区软件集团2011年度策略发布会召开

大会以“纳百川、行天下、融智慧”为主题，回顾和展望了IBM软件集团成立十五年以来“兼容开放与自身创新并举”的发展路线，明晰了IBM“智慧软件·智慧行业”的行业深耕战略，并再次强调了其“共同创造、一起分享”的渠道策略。

金蝶友商网新战略意在推动SaaS市场发展

3月17日，金蝶友商网发布最新战略，确立了“软件+服务+平台”发展模式。利用这一模式，友商网将提供“友商迎客宝”、“友商智慧记”等SaaS软件和友商KIS产品，为小企业提供财务、业务、客户营销管理、在线订货平台等关键应用；此外，友商网还将开展一系列增值服务；并通过开放平台，广泛联盟互联网企业等第三方服务，为企业提供一站式平台服务。

第四届“英特尔杯”全国大学生软件创新大赛开幕

3月11日，由教育部示范性软件学院建设工作办公室主办、云南大学软件学院承办的第四届“英特尔杯”全国大学生软件创新大赛暨MeeGo程序开发大赛正式拉开帷幕。这是大学生软件创新大赛连续第四年举办。本次大赛邀请了全国58所高校的在校本科生及研究生参加，广泛征集基于开源操作系统MeeGo的应用程序参赛作品。

2011年IBM Rational Jazz创新大赛开幕

3月14日，本届大赛正式启动。Jazz是面向软件交付技术的下一代协作平台，专门面向全球化和跨地域团队开发。与上一届不同，本次大赛参加人员扩大到两岸三地，而且不再局限于在校大学生，社会工作人员也会作为一个单独的项目，参加此次大赛。

ThoughtWorks技术雷达

(2011年 1月)

1月份最新发布的《技术雷达》认为以下几个技术趋势越来越明显：对软件持续交付及其实现方法的关注；云端产品的多样性和深度；更高效地运用基本的互联网技术。

ThoughtWorks技术咨询委员会由ThoughtWorks内部的资深技术专家组成，他们经常聚在一起讨论全球技术战略和对行业有巨大影响的技术趋势。这就促成了《技术雷达》的编制目的——帮助决策者了解目前能够影响市场的新兴技术和趋势。

《技术雷达》总结了这些讨论的结果，为包括从CIO到企业开发人员在内的相关人员提供有价值的信息。本文中仅提供内容摘要，读者可以

继续探索自己感兴趣的细节。我们尽量保持雷达的简洁明了，使读者能够快速了解。技术雷达用图解的形式，将所有项目分为技术、工具、语言和平台几大类。有些项目会同时属于多个类别，我们会将它归入最合适的某个类别。另外，根据这些项目目前所处的阶段，用不同的环将其进一步分组。这些阶段分为以下四个方面。

- **保留阶段**：该技术还在慎重的发展过程中。
- **评估阶段**：值得探索的技术，以了解其对公司的影响。
- **试用阶段**：值得应用的技术。公司有必要了解如何使用该技术，并且在风险可控的项目中试用。
- **应用阶段**：行业已经对该技术进行了试验，并找到了合适的使用方法；或者我们感到应用该技术的紧迫性，应该马上应用它，而不是逐渐部署。

详细介绍各类别时，我们将展示自上期《技术雷达》发布起，每项技术的动向。新的项目用三角形代表，上期已经提到的项目用圆形代表。在最近两期《技术雷达》中没有新动向的项目将不再列出，除非再有重大事件发生。

技术

企业全天候运行，但作出商业决策必需的信息，却还是在批处理模式下使用ETL作业，用过时的方法提供。随着企业转向全球市场，以及运营时间的延长，批处理的及时性更加无法满足要求。在企业做决策的那一刻，这些作业流程所提供的数据已经过时。在事务处理系统中，事件一发生，就应该被捕捉并送入数据

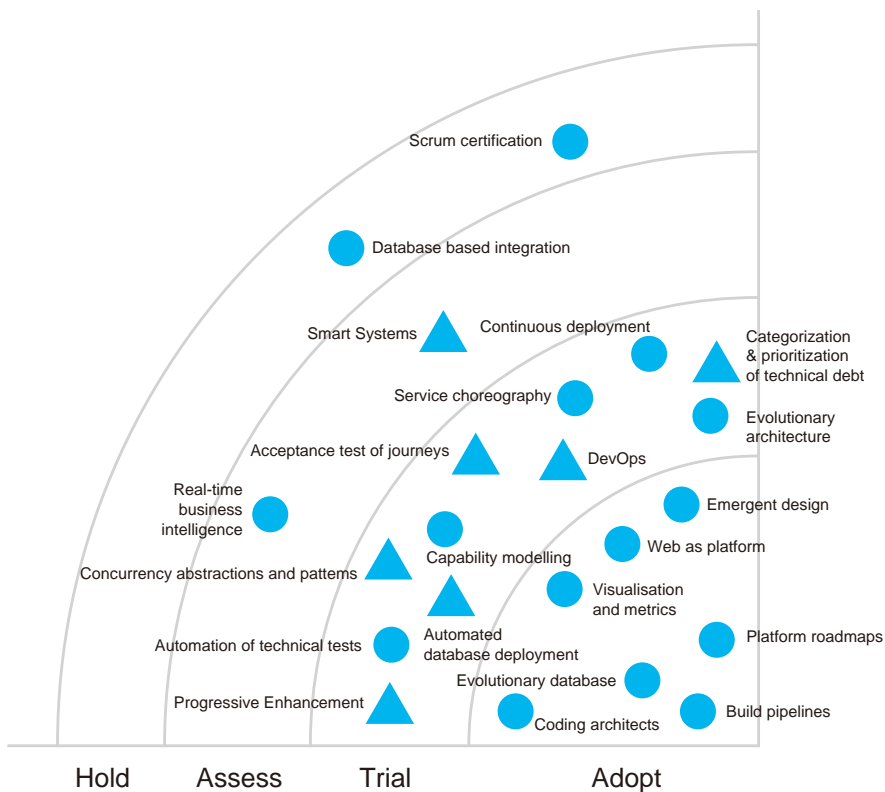


图1 “技术” 雷达图

仓库，这一点意义重大，可以让企业获得**实时商业智能（Real-time business intelligence）**。

不断涌现的智能系统（Smart Systems），融合了现实世界和虚拟世界，而配备有GPS、相机和大屏幕的智能手机只是其中一种。智能系统上可能的应用包括像Google Goggles一样的增强现实应用、位置服务和智能网格等。

由于开发和运维之间的脱节越来越受到重视，**DevOps**也继续受到关注。DevOps能够促进开发和运维之间的协同工作以及共同责任。DevOps为运维流程带来了更敏捷的实践，比如配置、变更管理和生产监控，也为开发带来了一类产品的思想、工具和环境。企业要实现应用程序的持续交付，DevOps是一个关键基础。

当成功完成向持续交付的转移后，数据库变化的部署也应该自动化，以便让依赖于这些变化的应用程序发布不必等待数据库变化的手动部署。**数据库部署自动化（Automated database deployment）**使得应用程序和数据库变化这个完整周期实现了自动化。

许多团队把重点放在story水平上的验收测试，但这会为维护测试带来很多困难，而且需要完成各个story而不是完整的功能。更加整体的考虑是将用户story组合成journey，然后对每个journey进行验收测试。一个系统中的journey是对用户和企业都有价值的一系列用户交互。起初，一个journey的验收测试（Acceptance test of journeys）将只包括一个步骤，但当story完成后，journey扩大到包含每个用户的进展。一旦对journey的验收测试通过，就意味着我们已经实现了真正的价值。

渐进增强（Progressive Enhancement）是一种网页设计策略，利用网络技术层，实现良好的用户体验。渐进增强重视可达性，允许使用任何浏览器访问基本内容。这种策略能使更多的内容被缓存，从而提高系统整体性能和可扩展性。

并发是一个越来越难以避开的难题。硬件的速度没有变快，但多核平台正在成为规范，连手机也开始应用双核或多核。**并发抽象和并发模式（Concurrency abstractions and patterns）**并不是新鲜事物，但不那么广为人知，它们可以帮助解决这一领域的许多挑战。特别是在Clojure、Erlang、Retlang和事件模式

中出现的模型，提供了一个比现有的线程、锁和信号量模型更具可测性和可靠性的方法。

对于我们构建软件时做出的妥协，技术债务是一个强大的、有效的比拟。不幸的是它已成为一个包罗万象的概念，包括了很多不同种类的问题和困难，从而导致了混乱和这个概念的“贬值”。处理这一问题的有效方法之一是对**技术债务的分类（Categorization of technical debt）**，用类似用户story的方式确定债务偿还的价值和优先次序。这有助于团队关注最重要的领域，并保持问题透明和可衡量。

工具

在现代数据中心或云部署中，存在大量的主机和设备，手动安装和配置这些基础设施是不明智的。**基础设施即代码（Infrastructure as code）**将基础设施的配置写成脚本或者用文件描述，并采用版本控制，配置的变化以受控的方式推送到数据中心。这和软件开发中使用的源代码控制和升级构建（build promotion）相类似，因此称为“基础设施即代码”。前端运行的两个基础设施自动化开源工具是Chef和

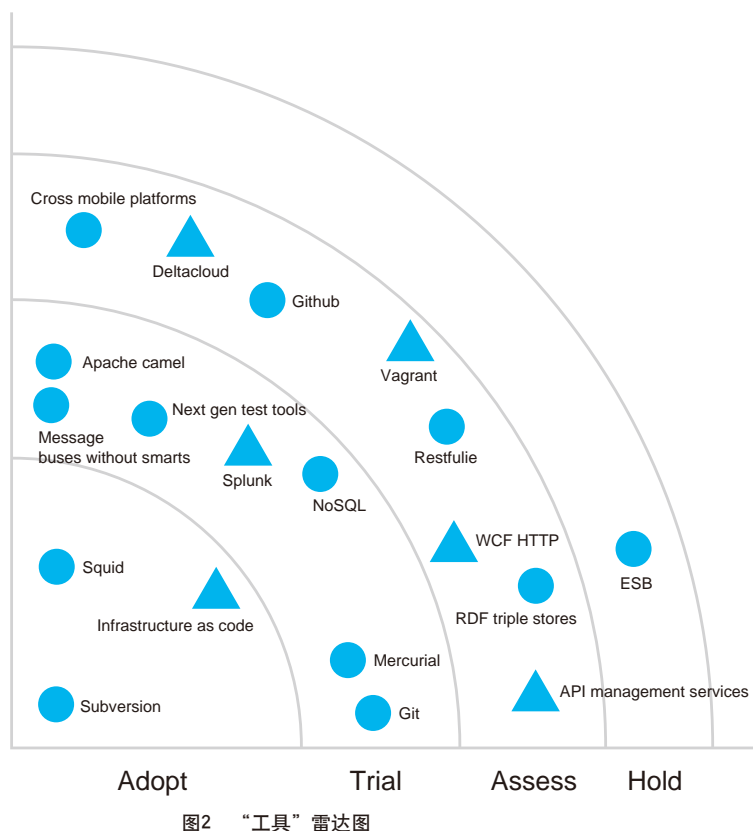


图2 “工具” 雷达图



图3 “语言”雷达图

Puppet。两者都使用textual DSL来编写自动化脚本。使用这种方法能实现一致且重复的环境改变，并能减少手动工作。在解决环境差异时，效果更加显著。

Vagrant工具能够方便地为团队分配使用便于版本控制的描述符构建的虚拟化开发环境。Vagrant有助于消除开发的环境差异，减少浪费在故障排除上的时间精力。

每个基础设施即服务（IaaS）云产品都提供了自己的API，用于执行常见任务。**Deltacloud**的目的是将这些API抽象化，并提供REST方式的接口，用于执行常见云管理功能，从而使得在云之间迁移虚拟基础设施变为可能。

不同的企业向庞大的开发社区公开API时，产生的非功能性需求通常都是一样的。密钥管理、认证、访问控制、流量管理、缓存、跟踪和报告都是常用功能，可以跨应用程序和公司重复使用，无需修改。一些服务提供商已经发现这种趋势，并通过“软件即服务”提供**API管理（API management）**。这个领域中，著名厂商包括Mashery和Apigee，它们都允许客户使用自己的基础设施部署服务。API管理服务也可能

会引起企业客户的兴趣，这些企业正在使用网络作为内部SOA的平台技术，作为更轻量级的选择，以替代传统的SOA管理工具。

微软的**WCF HTTP API**，尽管目前在发展初期，但发展很迅速，它对HTTP基本类型还有内容谈判和条件请求的惯例的支持已经让我们印象深刻。该API鼓励开发可测试性高、具有可明确分离的关注点的解决方案。让我们特别感兴趣的是该项目的开发方式，它立足于Codeplex上的开放源码社区。该社区引导.NET平台开发的能力曾经让这个项目进入之前的技术雷达。虽然现在的软件许可允许在产品中使用库，但面对目前该API的不确定性，不要在这个早期阶段依赖该API；在目前发布的版本中，我们欣赏的功能，比如说基于惯例优先原则的轻量级编程模型（attribute-light programming model），可能不会在正式的首发版本中出现。

应用程序日志既有好的一面，也有坏的一面。当产品出现问题时，我们可以求助于它，但实际上挖掘出有用的数据，通常需要用类似awk和sed的工具书写出代码，然后拼装在一起。**Splunk**是一个优秀的解决方案，能快速分析许多标准的日志文件格式，比如IIS、Log4J和syslog，并且可以扩展自定义格式。它可以静态或实时地索引文件，生成模板化或自定义的报告。如果原始日志中没有提供你所需要的字段，使用一个正则表达式，就能简单地内联或定义一个新的字段，以获得需要的细节。很难用语言描述Splunk的强大功能，所以我们建议你下载并试用一下。

语言

甲骨文公司收购了Sun以及它旗下的**Java**资产，为Java的未来带来了不确定因素。甲骨文有关Java发展蓝图的公告让人喜忧参半，继续了这种不确定性。因此，本期技术雷达继续强调这一问题。我们建议观察，而不是立即抛弃Java平台。

在以前的技术雷达中，我们将两个基于JVM的函数式编程语言——Clojure和Scala——归入评估阶段。我们曾经推崇过Clojure，因为这种语言更小、更集中。上期技术雷达我们也介绍了Scala更广泛的适用性，认为它更加适合于企业开发人员。我们还曾经目睹了采用Scala

后带来的巨大成功。因此，我们将Scala归入了试用类别。如果它被引入一个新团队，请特别注意Scala的习惯用法，以避免“没有分号的Java”或Perl的样式。

CSS已经成为样式化网页的一种非常受欢迎的和有效的DSL（Domain-Specific Languages，领域特定语言）。然而，它确实有一些局限性，因此，在CSS基础上诞生了一系列语言，让它更容易编写和修改。这些语言有SASS、SCSS和LESS，它们已经体现出了优势。

HAML语言允许我们使用缩进来布局HTML的结构。它不是普通的HTML替代品，它能够有效地聚焦标签的层次结构。

DSL是一种古老的技术，我们认为它的使用明显不足。我们希望Martin Fowler最新著作的出版，能鼓励更多人使用它。

平台

OpenStack是一款新型云操作系统，提供了完全开源的解决方案。它是结构云控制器，充分利用现有的KVM等虚拟化技术，并结合其他虚拟化工具，如Xen和OpenVZ。OpenStack目前快速发展中，预计在2011年第二季度末将提供一个稳定的生产就绪解决方案。

vFabric是VMware开发的新PaaS平台。建立在开源网络和信息平台Tomcat、Apache和RabbitMQ的增强版本的基础上，vFabric的目标是在多种云平台上建立一个基于Java的PaaS。目前支持的平台包括VMware和force.com合作建立的VMForce、Google的App Engine和Amazon的EC2。vFabric包括了GemFire公司的内存分布式数据管理平台和Hyperic公司的监控和管理工具，让它成为了Java开发者向云端移植时有用的技术集。

多年来JavaScript主要被用作客户端Web编程语言，但是作为一种轻量级语言，它可以很容易地嵌入不同的环境，包括服务器端。Node.js允许开发人员编写客户端和服务器的JavaScript应用程序。大多数服务器花费大部分的时间等待I/O操作，Node.js的事件驱动无阻塞架构则可以非常有效地解决这个问题。不同于基于线程的解决方案，在处理传入请求时，Node.js不需要等待I/O操作完成。实现高性能的服务时，Node.js是一个很好的选择。

Heroku是一个很简单的PaaS平台，支持

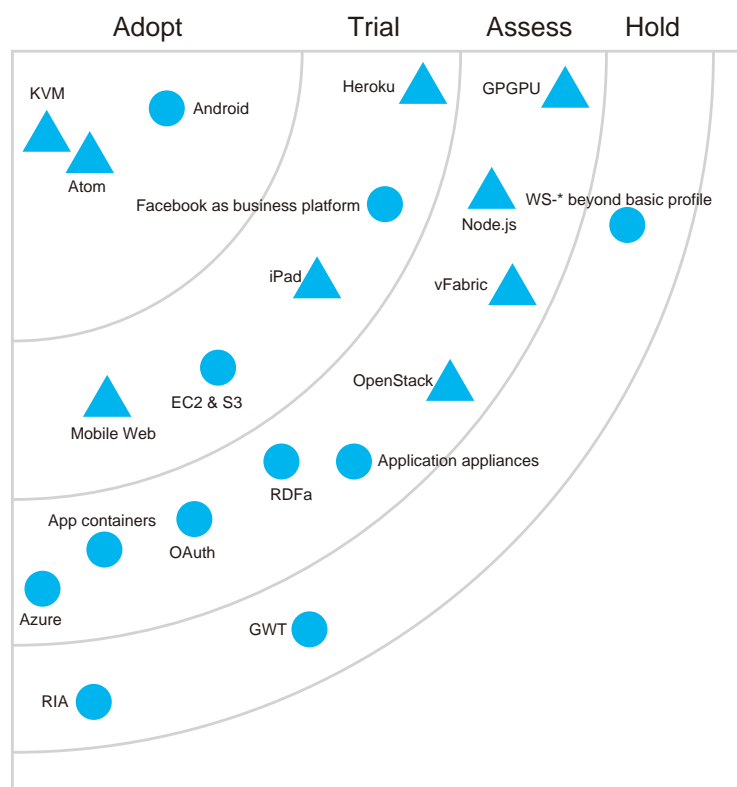


图4 “平台” 雷达图

Ruby on Rails这样的兼容框架。其他语言的PaaS平台，往往需要一个特定的编程模型，从而限制了开发。相对于同类产品，Heroku中使用标准的Rails堆栈，甚至允许配合普通的Git push进行开发。Heroku最近被Salesforce.com收购，它的服务获得了有力的支持。

移动网络（Mobile Web）在以前的技术雷达中处于评估阶段，但这次被分到试用阶段，因为通过iPhone、Android和平板电脑，移动网络的使用已经越来越多。许多设备可以提供近似桌面浏览器的经验，但为了向用户提供真正的最佳体验，我们建议，针对特殊移动设备的屏幕尺寸、显示和控制行为，对网站进行调整。渐进增强之类的技术，可以让一个网站同时适应桌面和移动浏览器。

大屏幕移动设备，如Apple的iPad和Amazon的Kindle，为无处不在的计算提供了新模式。其电池寿命长、接口简单、易于连接，有可能改变我们与计算机交互的方式。Apple的新用户界面抛弃了Mac电脑1984年引入并一直使用的文件和文件夹这些熟悉的桌面比拟（Desktop metaphor）。P



主持人：
冯大辉
现任丁香园（<http://www.dxy.cn>）网站CTO。曾历任支付宝架构师、数据库团队负责人等职。

架构师接龙

杨海朝 VS. 孙立

杨海朝：在错综复杂的庞大系统中，如何避免多条线频繁地发布新代码对线上业务的影响？

孙立：在公司或者业务快速发展的过程中，应用系统的功能越来越丰富，系统模块也变得非常繁杂。由于前期没有进行良好的整体架构，后期又为了快速响应需求，所以系统模块、多个业务线之间的耦合性非常强，很可能代码也不太规范，导致出现发布了A系统的A模块的代码后，B系统的B模块出现了问题，而B系统的B模块开发人员花了一整天的时间才发现问题是由于A系统的某个发布版本导致的。所以可能出现在系统长期运行稳定、压力不大的情况下，突然宕机或者报错，经过大量时间的诊断，发现是由于另外一个系统发布了新版本，接口不兼容所导致。解决这些问题，不仅会浪费开发人员的大量时间，而且这些问题对线上业务的影响也是巨大的，如何减少甚至避免这些问题的发生呢？我认为在整个系统架构上应该有这样的考虑。

代码版本回滚。代码回滚功能可以快速地使出现问题的错误版本轻松恢复到上一个正确版本。最简单的方式就是通过svn来实现。如果系统在发布新版本后，出现了没有预估到的错误或者故障，那么就可以快速回滚到上一个发布版本。谁也不能保证新发布的版本万无一失，所以快速回滚功能是需要频繁发布的系统所必需的功能。

分层接口化和组件化。接口化和组件化可以促进系统模块之间更加抽象化，在更换接口的实现方式或者组件时，不更改调用者的代码，就可以使系统稳定地平滑过渡。2010年，手机凤凰网就通过这种方式，在不更改代码、不停机的情况下平滑地将底层存储从TTSerVer迁移到自行开发的NoSQL存储

	提问嘉宾 杨海朝		回答嘉宾 孙立
<p>新浪首席 DBA，负责整个公司的数据库管理工作。热衷于数据库设计、性能优化、分布式部署方案和高可用性方面的研究。在大规模高并发、海量访问特别是大规模数据库运维方面有丰富的管理和维护经验。</p>		<p>去哪儿网高级系统架构师，曾就职于凤凰网、酷6和搜狐。对分布式搜索引擎开发、大数据量网站系统架构优化、运维监控等有丰富的经验。开源项目phplock和phpbuffer的作者，近期开发了NoSQL数据库存储INetDB。</p>	

上。接口化和组件化还能更好地实现代码重用，开发人员可以把接口实现类和组件开发得更加稳定。因此，在系统内部，要尽可能地分层接口化和组件化。

服务化。服务化应该算是对接口化和组件化一个更高的抽象，是将系统需要的某些功能独立成单独的、稳定的可靠性服务，通过约定的协议进行交互。比如系统的用户登录可以独立成SSO服务，用户IP归属地功能、文件上传功能、图片处理也都可以分别独立成单独的服务。我们可以部署一套服务进行抽象化，供所有的系统使用。这样服务和系统的功能都简化和抽象了，也更加容易维护和开发新功能。可以想象，在系统发布新版本后，就不会再影响上传功能、图片处理功能、登录功能的正常运行了。

自动化测试。自动化测试有助于发现人工不好发现的Bug。开发人员在开发程序时，尽量多使用单元测试；在代码发布之前，进行自动化的单元测试检测，只有全部单元测试通过后才允许发布。

应用监控。通过上面的方法并不能100%地避免由于代码频繁发

布带来的系统影响，但是我们需要尽可能快地知道带来的影响。所以我们需要对应用进行全面监控，这里的监控不同于系统运维人员使用的Cacti之类的对操作系统、数据库等底层的监控，而是对系统的输入输出、系统服务接口的进行正确性、响应时间等的监控。

杨海朝：在设计一个大规模伸缩性服务时需要做好哪些准备？是否会考虑分区、故障、自动化、异步等方面的问题？

孙立：在设计一个大规模系统时，我们首先要考虑满足用户需求，如果不能满足，那么这个系统就是废品。其次是系统的可伸缩性，我认为可伸缩性大致可以从以下两个方面来看。

功能和需求的伸缩性。功能和需求的伸缩性是指随着业务的发展，可以在现有系统上方便地增加、删除一些功能，并能快速应付需求变更。系统要想实现功能和需求的伸缩性，架构师不但要深刻理解需求，还要深入到业务中去，要对业务有深刻的理解，然后结合系统设计，考虑各种业务需求可能的变化和发展。

性能的伸缩性。性能的伸缩性是指系统能应付不断的数据量增加和访问量增长。性能的伸缩性可以通过硬件的横向扩展（Scale-Out）或纵向扩展（Scale-Up）来完成。通过升级硬件带来的纵向扩展虽然易于实现，但能力有限，所以在设计大规模系统时，必须采用通过增加机器的横向扩展方案。

要实现横向扩展，最简单的方式就是让系统无状态。比如Web服务，如果不使用本地文件、Session等有状态的模式，即可快速地在负载均衡后端增加机器进行扩容。然而，最终的数据库往往是集中式的，所以还得根据应用场景来进行数据的分区、主从复制等来实现横向扩展。

要保证一个大规模伸缩性服务具有高可靠性、高性能、高可用性，还需要进行一些故障的自动处理（比如故障的自动切换），进行全面的有效监控。面对一些跨机房请求、网络连接处理、I/O密集型、CPU密集型等操作时，我们要考虑是否需要异步化、线程池化、使用事件驱动模型、多种语言的融合使用等。

杨海朝：对基于网络的分布式应用，网络传输是一个影响应用性能的重要因素。如何

在架构中合适地使用缓存来节省网络传输带来的开销？

孙立：一个大规模可伸缩性应用，必然是一个基于网络的分布式应用，网络传输在其中扮演着重要的角色。按照网络传输的类型，大致可以分为同机房（局域网）和跨机房（广域网）两种网络传输。

同机房网络传输不但速度快，而且非常稳定。正因为如此，大部分开发人员不曾注意到同机房数据传输潜在的问题，即使是我们常用的千兆交换机、千兆网卡也是有千兆上限的，在高并发下，非常容易跑满上限，影响整体的性能。所以，在架构一个大规模的分布式应用时，需要适量考虑本地Cache、数据压缩传输等以节省内网网络传输量。比如Memcached服务器缓存了大量的大文本信息，那么最好使用压缩，这样不但可以节省网络传输，还能节省内存。

为了解决架构设计中不断出现的超过预估的数据量和访问量，需要做到以下方面：

- 第一，进行良好的可伸缩性设计；
 - 第二，要对数据量和访问量进行监控，并保持敏感性；
 - 第三，控制恶意访问。
-

跨机房网络传输的速度和稳定性都得不到很好的保障，而且带宽还是需要付费的。在国内环境下，跨运营商的网络传输就更加不稳定。如果应用程序通过阻塞的方式进行跨机房请求，在网络不稳定时，对系统的冲击将会是灾难性的。解决跨机房网络传输的方案大致可分为客户端和服务器端两种模式。

客户端模式主要是通过浏览器的跨域处理能力把不同机房的数据组合在一个页面。一般通过iframe、JS回调等方式进行跨机房数据传输，常见的第三方统计系统就是使用的这种方案。客户端模式不但容易实现，而且避免了对服务器端产生额外压力，通过异步化处理，还能提高页面渲染的速度。不过还需要考虑到客户端模式安全性低、不能处理复杂业务逻辑、不支持WAP网站等问题。

服务器端模式处理跨机房网络传输要复

杂很多，需要考虑数据一致性、避免阻塞式网络传输影响性能、带宽成本等。一个高并发应用，如果存在大量的阻塞式跨机房请求，不但占用大量的带宽成本，而且性能和访问速度也会急剧下降。根据用户在一定的时间段内上网地点固定不变的原理，那么用户始终都是访问的同一个机房。这样就可以针对用户来做Local Cache，缓解跨机房传输的成本。机房之间通过队列的方式进行异步和压缩传输，不但能极大地提高系统性能，而且对用户来说，几乎也感觉不到延迟。

杨海朝：在架构设计中如何应对不断出现的超过预估的数据量和访问量？

孙立：系统在实际的运营过程中，虽然经过测试和调研分析了可能的数据量和访问量，但还是有可能因为突发事件、预估不准确而导致数据量和访问量大大超出预期，那么我们一般可以从以下几个方面来解决。

进行良好的可伸缩性设计。出现超过预期的数据量和访问量时，能够快速通过增加机器进行横向扩展，而无需改写和优化系统代码。一个无状态的应用是最容易扩展的。Web服务器可以很容易地设计成无状态模型，从而可以方便地增加机器进行扩展。数据库和存储可以通过数据分区来进行扩展，数据分区其实也算是对状态的弱化。

要对数据量和访问量进行监控，并保持敏感性。监控能使我们更快地了解系统数据量和访问量增长的情况，甚至可以预测短期内可能的数据量和访问量增长，以便提前进行扩容准备。

控制恶意访问。对恶意攻击性访问，我们必须进行阻止，以防止影响正常的系统访问。比如一些不知名的爬虫恶意爬取、接口调用方由于Bug导致了大量的访问，所以在操作系统层和应用层都应该对恶意访问有所控制。

杨海朝：分布式的数据存储层如何结合应用特点选择合适的存储方式？比如MySQL、Redis、MongoDB等的应用场景。

孙立：选择存储方式首先要考虑满足应用的功能需求，其次要满足应用的伸缩性需求，最后需要考虑的是性能，结合具体的应用场景再进行权衡选择。

Web2.0时代具有大数据量、高并发等特点，所以我们往往会结合应用的特点，把多种

数据存储结合起来使用，使它们发挥各自的优点。比如复杂点的关系查询考虑用MySQL，高并发的点击统计考虑用TTSer，列表类型的缓存考虑用Redis，字段不固定的文档存储考虑MongoDB。同一份数据还可以存入到多种存储，根据查询场景分别选择适合的存储查询。

杨海朝：分布式数据库中如何考虑一致性、分区策略和延时问题？

孙立：在系统架构中使用分布式数据库时，我们必须面对一致性、分区策略和延时问题的处理，也就是著名的CAP理论阐述的问题。对CAP的选择需要结合我们的应用场景来选择。

一致性问题。其实延时问题也是一致性问题。一致性可以分为强一致性、弱一致性和最终一致性，并不是所有的应用都需要强一致性。根据应用的特点，为了提高系统的整体性能，我们还可以考虑会话（Session）一致性。比如用户发表的微博、评论数据，用户自己可以马上看到，但是其他用户可能需要等一段时间。还有一些场景需要强一致性，比如支付问题。

MySQL数据库一般会使用Master-Slave复制模式进行扩展。在高压力访问情况下，MySQL数据库主从复制延迟可能变得非常高。我们可以从上面提过的会话一致性来解决复制延迟问题。另外一个方案就是提高复制速度。在MySQL主从复制过程中，从服务器解析出bin-log日志后，向从数据库执行写入操作是单线程的。如果你可以修改源代码把MySQL复制变成多线程那就更好了，还可以通过增加从库实例，从库只复制部分表的方式来变相地进行并行复制。MySQL5.1以上版本提供的行复制模式也可以加快复制速度。

分区策略问题。为了提高数据库的性能和可扩展性，我们往往会根据某些规则把数据进行拆分存储。分区策略应该遵循查询能在一个表完成的原则，尽量避免跨表查询。不管以Hash、Range或者其他方式都必须遵循这个原则。如果因为有多种关系查询，无法达到一次分区就满足上述原则，那么可以进行多次分区和冗余存储。📌

产品管理与项目管理、产品设计的关系

文 / Marty Cagan, 译 / 孙洋 林云源

Marty Cagan是享有世界声誉的产品管理专家，曾经担任网景副总裁、eBay产品管理及设计高级副总裁。本文是他回顾自己二十多年来从事软件产品管理工作的总结和经验分享，谈到了产品管理、项目管理、产品设计的关系。

产品管理与项目管理

上期文章我阐述了分清产品管理职责和产品营销职责的重要性。许多公司遇到另一个问题：产品管理和项目管理的复叠。

许多互联网公司沿用了开发零售软件的惯例，用产品管理涵盖了项目管理。在传统的零售软件领域，产品经理常常兼任项目经理的工作（例如微软最出名的Office系列产品就是采用这种模式开发的）。这种模式虽然适用于单独发布的软件产品，但不太适合开发互联网服务类产品。

为了解释个中缘由，我先简单回顾一下互联网产品的历史。互联网服务类产品大约出现于1996年，当时我们曾“纠结”是否继续称自己为产品经理，毕竟像旅游网站这类产品不是传统的零售软件，更像是面向客户的服务。不过这种困惑很快就被人们抛到脑后了。

早期的互联网公司让产品经理继续兼任项目经理的工作，网景和雅虎就是这样，但他们很快就遇到了问题。在零售软件领域，产品通常以独立安装包的形式发布，发布间隔从几个月到几年不等，产品和项目具有相同的粒度，开发频率也相同，产品经理兼任项目经理相对容易。但是在互联网产品领域，这种模式行不通了。

互联网服务类产品对整站代码的局部修改更加频繁，发布周期明显缩短（通常是每周或每月发布一次），大部分项目的开发周期明显长于发布周期。为了适应这种变化，很快出现

了并行开发和火车模型发布模式。许多成熟的互联网公司都在使用这种发布模式。

火车模型本身是一个很大的课题。我想强调的是：每次发布都需要强有力的、有效的项目管理，它不局限于具体项目，必须统筹全局。每次发布可能包含多位产品经理负责的功能，要协调发布管理、程序开发、网站运维、客户服务、产品管理各个方面。有些互联网公司把负责火车发布的项目经理称作**列车指挥员**。

如果采用火车模型，指派项目经理来控制产品发布，就不需要产品经理兼任项目管理工作了。

再回到互联网历史。随着雅虎、网景、AOL等公司的产品线和网站不断发展，产品发布流程变得越来越复杂，项目管理职能慢慢从产品管理中分离出来。这些公司逐渐具备了专业的项目管理能力。许多后起之秀，像eBay和Google，如果不是拥有强有力的、能够统筹产品管理、程序开发和网站运维的优秀项目管理团队，不可能开发出那么多高质量的产品。

就互联网公司而言，把这两种职责分开至关重要。必须持续不断地推动项目，否则产品发布就会延期。

就传统的零售软件而言，我认为也有必要把这两种职责分开。这与产品管理的性质有关，产品管理的职责是发现有价值的、可用的、可行的产品；而项目管理则关注于如何执行计划按期交付产品。



Marty Cagan

过去20年，Marty Cagan作为负责定义和开发产品的高级经理人为多家一流企业工作过，包括惠普、网景通信、美国在线、eBay。他亲历了个人电脑、互联网、电子商务的起落沉浮，致力于通过写作、演讲、培训帮助客户打造富有创意的产品。为此，他撰写了《Inspired: How to Create Products Customers Love》一书，创办了硅谷产品集团公司（SVPG）。在此之前，他的最后一份工作是担任eBay产品管理及设计高级副总裁，负责规划全球电子商务网站的产品和服务。

怎样成为优秀的项目经理？

在所有成功的企业里，都可以找到这样一群人，他们是真正的精英，是他们让公司变得与众不同：优秀的产品规划、持续的商务拓展、争取更多客户、准时交付产品；而普通公司的表现恰恰相反：糟糕的产品规划、眼睁睁看着客户流失、产品发布不断推迟。

eBay就属于这类成功的企业，公司各个部门都拥有最优秀的人才。eBay有着独特的产品开发流程，突出表现在三个方面：超高的生产效率、严格的标准、基于强力项目管理的可预测流程。

琳·里迪（Lynn Reedy）是我在eBay的同事，她是我见过的最出色的项目管理人才。她是eBay项目管理的中流砥柱，我很荣幸曾与她共事。加入eBay之前我曾自嘲是项目管理的行家里手，她向我展示了真正的项目管理能力。

在一些大公司（例如微软），产品经理也负责项目管理工作。我认为，强有力的项目管理能力是产品经理的优势。至少你可以使产品更快进入市场，甚至很可能会决定产品能否面市。但我还是认为产品经理和项目经理应该是两个独立的职位。

有些项目经理以为管理能力等同于使用微软Project软件的能力，他们没有领悟项目管理的真谛。以下是我从琳·里迪这样优秀的产品经理身上总结出的七个特点。

紧迫感。只要琳走进房间，立刻就能传达给大家一种紧迫感。大约60秒闲话开场白后，马上转入正题。这种效果表面看来是因为她独特的身体语言和气质，但事实上，紧迫感和高效率是eBay企业文化的核心，而且已经升华为琳人格的一部分。

捕捉问题的能力。有无数原因导致会议效率低下、毫无建设性；其中最主要的原因是会议目的不明确，不清楚要解决什么问题，也不知道难点在哪。优秀的项目经理能够迅速、准确地指出问题及其要害，改善会议效果。

思路清晰。引发典型业务问题的原因多种多样，比如政治因素、日程安排有冲突、同事个性不合等，如果置之不理，稍作拖延就会导致整个项目一团糟。项目经理需要排除感情因素，放下思想包袱，拨云见月，把待解的问题逐一独立分离出来，分配给每个同事专注解决。

用数据说话。优秀的项目经理明白数据的重要性，懂得利用数据识别项目方向，确认项目进度。他们知道改善产品和开发流程，必须从测量、收集数据开始。在项目进度紧迫的状况下，最容易仅凭直觉草率行事，为避免这种情况，项目经理务必坚持根据数据和事实做出决策。

果断。在多数公司里，产品团队成员不必向项目经理汇报工作，但项目经理必须驱动同事做出决策。项目经理必须向大家传达一种紧迫感，及时向团队收集数据和建议，适时向上级部门汇报情况，把问题理顺，用理性的思路和清晰的理由，帮助大家用数据做出决策。

判断力。以上这些特点都基于良好的判断力。项目经理必须清楚何时催促进度，何时向上级汇报，何时需要收集更多信息，何时找个别成员私下交流。判断力很难言传身教，只能靠自己积累经验。

态度。如果产品不能按时交付，我们总能听到各种理由——可行性太差、资源不足、时间不够、资金匮乏，等等。项目经理绝不能为自己找借口，必须克服所有障碍，解决所有问题，一往无前、愈挫愈奋，直到梦想成真。

我相信如果不是琳带来的卓越的项目管理方法和企业文化，eBay绝不会有今天的成就。

产品管理与产品设计

很多产品经理向我抱怨他们公司不懂用户体验设计，公司的用户界面设计师因循守旧，没人负责这方面的工作。他们不得不分散精力亲自参与设计，即便如此，产品的用户体验还是很糟糕。有些公司直到产品开发后期才仓促把视觉设计外包出去，只为赶在质检前给产品穿上漂亮的外衣。

还有些人说他们公司压根不知道用户体验设计是什么，不明白设计对产品有多么重要。很多公司都没有意识到这个问题的严重性。

在我看来，这是因为设计界未能充分向外界传达一种理念，即设计对产品而言相当重要。虽然设计界内部沟通很流畅，而且拥有一批杰出的设计师，比如马克·赫斯特（Mark Hurst）、休·杜伯里（Hugh Dubberly）、艾伦·库珀（Alan Cooper）等，但他们很少向外界普及他们的想法。相比之下，缺少设计师的团队

更需要这些信息。设计界应该向所有产品团队（特别是产品经理）宣传产品设计的重要性和价值。

我重视这个问题的原因很简单，好产品必须提供舒适的用户体验。舒适的用户体验是产品管理和用户体验设计共同作用的结果。

这是个很大的话题，我们先从了解设计包括哪些角色分工开始。这里我给出与用户体验设计密切相关的分工。注意我描述的是工作角色而不是个人，因为有的人可能承担多份工作。这些角色都是设计舒适用户体验所必需的。

用户研究：专门研究、分析用户，评估产品或产品原型是否符合特定用户的使用习惯。包括拟定恰当的测试项目、监督测试、评估测试结果、提出改进方案。

交互设计：在理解目标用户的基础上设计有价值的、可用的目标功能、用户导航和产品使用流程。交互设计师通常用线框绘制产品需求，然后交给视觉设计师。

视觉设计：根据线框设计可见的用户界面（页面），包括严格的布局、颜色和字体设置等。视觉设计能够传达并唤起产品蕴含的情感（其重要性常常被低估）。

迅速制作原型：开发融合了产品经理和设计师创意的产品原型，让用户试用，并根据反馈反复修正原型。

以上四种角色与产品经理密切合作，将功能与设计相结合，满足用户需求。目标是确保产品同时具有可用性（用户明白如何使用产品）和吸引力（用户对产品的渴求程度）。

除此以外，还要确认软件设计是切实可行的，因此必须让软件架构师评估设计进展和产品原型。这部分内容稍后再详细介绍。

对大型产品（尤其是针对消费者的互联网服务）来说，这四种角色缺一不可。开发企业级应用软件的公司如果想从众多竞争对手中脱颖而出，最简单的办法是提供优秀的用户体验。用户体验是大部分企业级产品的弱项。

对小型产品来说，可以让一位设计师身兼多职。例如，我最近与一家创业公司合作，开发针对消费者的Web2.0服务。对方只有三个人：一位产品经理、一位交互设计师（同时负责用户研究）、一位视觉设计师（同时负责开发原型）。他们的工作非常出色，很快就拿出

了可供目标用户测试的产品原型。

还要提醒大家一点。很多公司希望改善产品的用户体验，把用户体验设计外包给设计公司。这在一定程度上是可行的，但是有些工作不适合外包。例如，我认为交互设计不能外包，有三个原因：

- 深入理解用户和客户的需求非常费时间，需要多个项目的实践积累。设计公司没时间深入了解客户需求，就算他们做到了，这些经验也很难保存下来，用到下一个版本里。

- 交互设计师必须现场深度参与项目开发，从立项直到产品发布。开发和测试过程中会出现各种细节问题，必须有一名交互设计师迅速做出决定。

- 产品的用户体验是公司的核心竞争力，必须在内部完成。如果让我选择，质量检验更适合外包。

只要团队中有一位称职的交互设计师，视觉设计也可以外包，毕竟视觉设计公司很多，完全可以满足你的需求。此外，用户研究和可用性测试也可以外包，只是成本较高，对我这种重视测试反馈的人来说更是如此，所以我建议让产品经理和交互设计师分担这项工作。

至于原型开发，你可以从开发团队借个帮手。要让他明白原型代码和产品代码毫无关系，原型代码绝不能用在今后的实际产品里。最好直截了当地告诉他，原型的所有代码都是要废弃的。

这个话题涉及的范围很广，限于篇幅，讨论暂告一段落。但我希望以上介绍能为今后的讨论奠定基础。你的团队有人负责这些工作吗，还有哪些工作被忽略了？🔴



本文节选自华中科技大学出版社《启示录：打造用户喜爱的产品》一书。该书从人员、流程、产品三个角度介绍了现代软件（互联网）产品管理的实践经验和理念。特此感谢华中科技大学出版社与Marty Cagan先生授权。

互联网创业团队环境文化建设谈

环境是最忠实的执行者



王建硕

百姓网CEO

把水龙头稍微松开一点点，让水可以每分钟滴一滴，放个桶在下面，然后忘记这件事情（不用我提醒，你一定会忘的）。等你再过来看的时候，或许桶早已经满了。这个实验成功的前提是，你一定要忘记你做了这件事情，只要你还惦记着，桶里面的水就不会太多。

企业文化也是一样。真正有用的举措，一定要做一些像滴水这样的事情，在你忘记了它存在的时候，它还依然忠实地执行它的职责。

所以说，仅仅落在书面上的规章制度是没有用的，开大会也是收效甚微的。这些变革，无论当时多轰轰烈烈，要不了一个月，就在时间的冲刷下消散得无影无踪。

而像海报这样的东西则不同。贴上去的时候不见得有人能注意到；贴了三个月，你会发现真的会有作用，因为它每天都在对周围的人有些影响。而这三个月里面，可能你的注意力早已经转移了不知道多少次，也必然忘记了什么时间把它挂上去的，而它就在你忘记的时候，帮你提醒着自己和周围的人。

改变人的行为的最好办法是通过改变环境。因为环境是最忠实的执行者，你做了改变，它就一直在哪里；你已经忘记了，它不会忘记。


这是一些从我们的办公室布置中获取的一些点滴灵感。

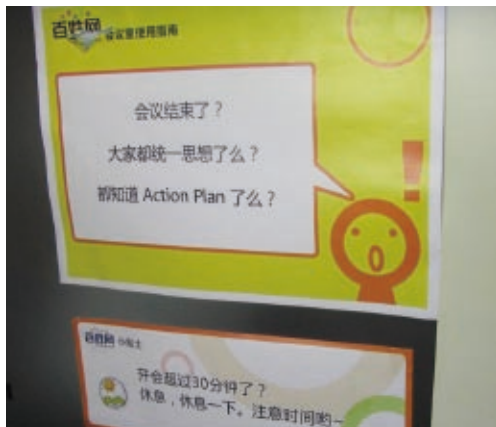
- 如果沟通不足的话，就多放些沙发。
- 如果一个地方特别容易滋生闲谈，就去掉那里的椅子，或者干脆放几盆植物。
- 如果开会容易超时，就放一个大挂钟。
- 可以写字的玻璃墙旁边，一定要放笔。
- 同一个团队一定要坐在没有隔板的大房间里，无论有多挤。



同一团队要坐在没有隔板的大房间

- 不同团队之间，要加上隔板以避免干扰。
- 办公室只留一个进出口，让团队成员有机会通过一个走廊，以增加大家打照面的机会。
- 饮水机旁边一定要有沙发。
- 如果一件事情足够重要，印在T恤衫上面分发给团队。
- 没有什么比免费的水果更能把大家围拢在一起。

● 技术团队要有一面墙画系统架构图、拓扑图，甚至核心代码——除非把大家在做的东西形象地展示出来，否则千万不要以为所有人都知道他们在做些什么……



会议室个性化提醒



由细节引发的企业文化思考



于敦德

途牛旅游网CEO

一些公司往往过于关注战略，很少去关心细节问题。创业这些年，我对于细节深有体会，尤其是员工越来越多，细节就越显重要。我们内部办公都是用邮件来沟通，有一个细节是一封邮件发出去后，大家收到后总会回复一下，如“收到”、“好的”之类。虽然只是一个小细节，但导致的结果是大家邮箱里面每天都会收到很多类似“收到”、“好的”、“谢谢”之类的邮件。这些无实际意义的邮件往往会对工作造成一定的干扰，所以我们就规定，除审批回复外，如果回复邮件里面仅有类似感情用语或者表示得知，则不需要发送。我们默认所有人都会阅读收到的邮件，如果因为没有阅读出现问题，默认是收件人的责任。这虽是个小细节但保证了我们邮件沟通的高效性。

其实细节就像是润滑剂，让公司能够顺畅地运转，同时还能对公司这个机器起到保养作用。办公室多一些绿色植物，每逢节日办公室做些氛围布置，这些都像是润物细无声一样，对公司企业文化起到很大的作用。作为企业文化细节不仅体现在公司内部管理上，对业务也同样发挥很大的作用。

我们有一个重要的文化叫“没有惊讶”，要求所有途牛人在与同事、客户的沟通方面做到细致入微，多从他人的角度去思考问题和解决问题，不要造成相互之间的“惊讶”，尤其我们作为服务性行业，细节更是凸显重要。公司一直强调，对于我们会员的沟通要注重细节，所有行程线路细节要尽量全部呈现给会员，不能让会员在行程中产生“惊讶”。

牛人堂：企业的学校，员工的课堂

我们在新员工培训的时候，会向每一位新员工传输一个理念：“每一位途牛人都是牛人！”牛人堂是员工学习的课堂，同时也是进行自我提升的一个优化空间。

一家企业要想把自己的文化传播给自己的员工，首先就是要有完善的培训体系，让自己的员工在企业里不断学习和进步。牛人堂就是我们途牛自己的员工学校。在这里，有系统的新员工入职培训，有专业的培训讲师授课；有老员工的业务交流，相互学习好的经验；有专业技能的学习课堂，提高员工的业务流程效率；有相关培训专家座客，教授最实用的职场培训……在各项培训中，培训部还会在培训中开展很多符合年轻人口味的趣味性活动，以达到良好消化知识的目的。

同时，每位员工入职一个月内就会有和公司高层直接对话的机会，我们称之为新员工沟通会。新人可以和公司高层面对面沟通，主动提出疑问和见解。

员工TOP排行榜：提升内部竞争力

企业的发展源于员工的创造力和竞争力，需要员工的集体力量来推动，员工的个人发展则需要企业提供机遇和发展空间。对于企业而言，有效的绩效考核机制会提高整个公司的员工业务能力；对于企业员工而言，有竞争才会有进步。我们在每个业务部门办公区域设置了一个本部门的TOP排行榜，定期表彰工作积极的员工，鼓励其再接再厉；同时对其他员工也起到激励作用，形成良性的内部竞争机制，从而使员工不断进步，带动整个企业的发展，提



员工TOP排行榜

一分钟先生

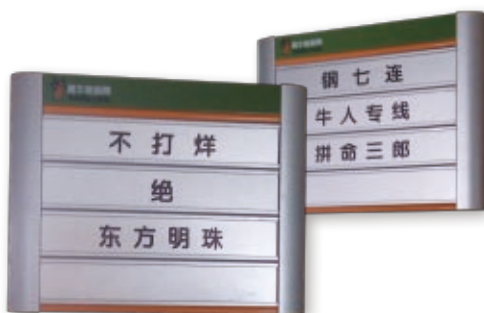
Mr. One Minute

高企业的综合竞争力。

门牌取名：团队最真实的自我写照

我们给员工创造的是一个轻松、活跃的工作氛围，以利于员工最大限度地发挥自己的创造力和工作潜能。途牛产品部的每个团队都有自己的团队名字和精神面貌口号：或是反映团队的工作职能，或是对自己团队的真实写照，或是团队自己立下的工作目标等。

“牛人专线”小组负责我们的主打产品线——牛人专线；“钢七连”、“拼命三郎”是团队的工作精神面貌和态度；“东方明珠”



有趣的门牌取名

是团队负责的旅游方向。这是每个团队员工集思广益，结合团队的各方面特色去总结的，是团队最真实的自我写照。自己给自己定位，自己给自己定目标好过于直接的上级对下级。

我们的会议室、会客室等命名也是结合我们公司的特色来命名的，不是用“大”、“小”会议室等来区分，而是“巴厘岛”、“可可西里”、“马来西亚”……进入公司一看就能多少猜到我们公司的行业了。

节日装点：增强员工的归属感

企业的竞争最终取决于人的竞争，为发挥出员工的最大潜能，除了给员工提供业务培训、提高员工待遇、培养员工的个人兴趣、组建足球俱乐部等兴趣小组开展业余活动外，给员工创造良好的企业归属感也是很重要的。在我们这个大家庭里，员工就是家人的观念深入人心，而员工拥有强烈的归属感是需要公司去营造的，所以在平时的节日期间公司行政部就会做一些温馨的装饰，给员工营造一种亲切的氛围，让大家感受到公司就是家。P

虾米网创业公司环境的思考



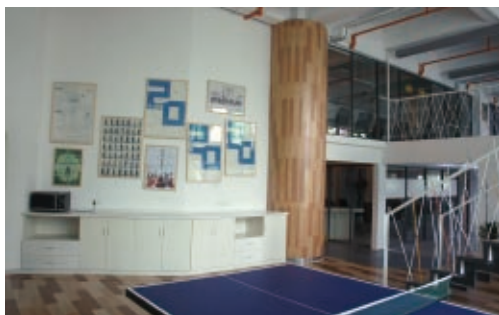
王皓

虾米网总经理

虾米网从创业到现在搬过三次家。第一个办公室在古荡的龙都大厦，商住两用楼，70平方米的住宅。之前的租户也是拿来开公司的，所以我们也没怎么特别装修，只是把桌子椅子搬进去就办公了。服务器就放在阳台上，所以耳朵里总是能听到服务器风扇的轰鸣。这儿唯一的好处是开窗就能看到对面的小和山，一年四季满眼绿意，适合放松。第二个办公室在枫华府第，毛坯房、跃层。简单粉刷了一下，从地板到墙壁都是一个糙劲。开始时编辑部和软件部分别一个房间，网站开发、运营、

设计部门共享了客厅，楼上还能有一个大厅做会议室。可没多久空间就不够用，于是大会议室被占用，客厅里连转身都困难。其实我想说的是，对于任何一家初创期的公司来说，办公环境之类的都是浮云，将有限的预算尽量花在设备和人工上会更好。而且，拥挤的环境和空间反而会拉近大家的距离。不过也不能说完全没有装饰，那时员工身边的墙都可以随便布置，有在上面画画的，也有贴照片的，从一楼到二楼的楼梯间墙壁上贴的是虾米网活动的大幅海报以及公司活动的照片，还有员工的一些

关注IT人自我管理、职业发展，广邀经验人士排忧解难。
花一分钟阅读，还你一个明白。
有难题吗？一分钟先生回答你！



富有创业气息的环境设计

愿望，比如希望周游世界等。

现在的办公室终于宽敞了，是原先的新华书店仓库，层高不错，适合做成Loft，好像这样比较有创业的感觉。当然，在装修前我也不太清楚最后的效果，这里要感谢帮做设计的朋友，没有把虾米办公室变成一个纯粹的办公室。


虾米网作为一家音乐网站，一直确信自己是网络行业里最懂音乐的，所以音乐的元素当然不能少。前台的背景其实是跳动的频谱，第一眼就有音乐的气氛。办公室整体色调和虾米网的颜色是一致的：橙色、白色还有深灰色，最大面积的是白色，支撑楼板的钢架我们刷成橙色，日光灯罩深灰色。另外，我想虾米最不缺的就是各种各样的海报。这些年虾米参与的线下演出活动加上自己组织的活动少说几十场，而且大部分海报还都有歌手的签名，所以我们至少省下了不少打印费，基本上在我们每个办公区域都可以看到有趣的海报和歌手对虾米网的祝福。

我们还在公司内安排了一个休息区，这里没有隔层，高度是5.8米，员工可以在沙发上休息或者打乒乓球，等到公司开一些全员会议时又能当作会议室用。

比较特别的地方是我们的音乐走廊。走廊是整个园区公用的，本来我们的办公室都是玻璃墙，私密性不太好，可如果直接用窗帘感觉又太单调了。于是我们讨论出一个有趣的创意，员工当模特，拍了各个年代的音乐设备发

展史装饰在墙上，如果有人从园区入口进来，按顺序正好可以看到音乐设备是如何从留声机一步一步发展到现在的，既美观又私密还很有我们虾米的特色。除此之外，我想虾米网应该是国内所有公司里拥有吉他最多的公司，现在光办公室里就有12把吉他，我们偶尔也会在公司里举办内部的吉他培训，员工无聊了随时可以抄起桌上的吉他玩一下。

在布置方面，我们也做了一些思考。比如在一楼的办公桌边上特地安装了白板，方便员工就地快速展开讨论，甚至在办公室的玻璃上也贴上不干胶，好当白板用。

不过，我更想说的是所谓创业气氛真正来自人，外在的东西只有在合适的人身上才能起到辅助作用。比如IT行业谁不希望能有Google或者微软那样的办公环境呢？但我相信即便预算不是问题，这样的环境也未必适合国内公司，我们的员工与创业者无论心态还是个人习惯上不一定就能接受这样的设计，看看我们的大学美国的大学你就知道差别在哪里了。再比如我们的办公桌是没有设置隔断的，我们希望员工可以更方便地沟通，而不是把自己孤立起来。但问题是，真正的隔断恐怕是在人的心里，如果他对创业的目标、愿景不认同，他怎么会轻易地与同事真正沟通呢？



虾米网员工自制的音乐发展墙

梦寐以求的编程语言(上)

文 / Paul Graham, 译 / 阮一峰

Paul Graham是公认的“硅谷创业之父”，对于许多技术问题，他有自己独到的见解。在本文中，他分享了何谓理想的编程语言。文章分两期刊登，第二部分将刊登于5月刊。

我的朋友曾对一位著名的操作系统专家说他想要设计一种真正优秀的编程语言。那位专家回答，这是浪费时间，优秀的语言不一定会被市场接受，很可能无人使用，因为语言的流行不取决于它本身。至少，那位专家设计的语言就遭遇到了这种情况。

那么，语言的流行到底取决于什么因素呢？流行的语言是否真的值得流行呢？还有必要尝试设计一种更好的语言吗？如果有必要的话，怎样才能做到这一点呢？

为了找到这些问题的答案，我想我们可以观察黑客，了解他们使用什么语言。编程语言本来就是为了满足黑客的需要而产生的，当且仅当黑客喜欢一种语言时，这种语言才能成为合格的编程语言，而不是被当做“指称语义”（denotational semantics）或者编译器设计。

流行的秘诀

没错，大多数人选择某一种编程语言，不是因为这种语言有什么独特的特点，而是因为听说其他人使用这种语言。但是我认为，外界因素对于编程语言的流行其实并没有想象中那么大的影响力。我倒是觉得，问题出在对于什么是优秀编程语言，黑客的看法与大多数的语言设计者不一样。

黑客的看法其实比语言设计者的更重要。编程语言不是数学定理，而是一种工具，为了便于使用，它们才被设计出来。所以，设计编程语言的时候必须考虑到人类的长处和短处，

就像设计鞋子的时候必须符合人类的脚型。如果鞋子穿上去不舒服，无论它的外形多么优美，多么像一件艺术品，你也只能把它当做一双坏鞋。

大多数程序员也许无法分辨语言的好坏。但是，这不代表优秀的编程语言会被埋没，专家级黑客一眼就能认出它们，并且会拿来使用。虽然他们人数很少，但就是这样一小群人写出了人类所有的优秀软件。他们有着巨大的影响力，他们使用什么语言，其他程序员往往就会跟着使用。老实说，很多时候这种影响力更像是一种命令，对于其他程序员来说，专家级黑客就像自己的老板或导师，他们说哪种语言好用，自己就会乖乖地跟进。

专家级黑客的看法不是决定一种语言流行程度的唯一因素，某些古老的软件（Fortran和Cobol的情况）和铺天盖地的广告宣传（Ada和Java的情况）也会起到作用。但是，我认为从长期来看，专家级黑客的看法是最重要的因素。只要有了达到“临界数量”（critical mass）的最初用户和足够长的时间，一种语言就可能会达到应有的流行程度。而流行本身又会使得这种优秀的语言更加优秀，进一步拉大它与平庸语言之间的好坏差异，因为使用者的反馈总是会导致语言的改进。你可以想一下，所有流行的编程语言从诞生至今的变化有多大。Perl和Fortran是极端的例子，但是甚至就连Lisp都发生了很大的变化。

所以，即使不考虑语言本身的优秀是否

能带动流行，我想单单流行本身就肯定会使得这种语言变得更好，只有流行才会让它保持优秀。编程语言的最高境界一直在发展之中。虽然语言的核心功能就像大海的深处，很少有变化，但是函数库和开发环境之类的东西就像大海的表面，一直在汹涌澎湃。

当然，黑客必须先知道这种语言，才可能去用它。他们怎么才能知道呢？就是从其他黑客那里。所以不管怎样，一开始必须有一群黑客使用这种语言，然后其他人才会知道它。我不知道“一群”的最小数量是多少，多少个黑客才算达到“临界数量”呢？如果让我猜，我会说20人。如果一种语言有20个独立用户，就意味着有20个人是自主决定使用这种语言的，我觉得这就说明这种语言真的有点优点。

达到这一步并非易事。如果说用户数从0到20比从20到1000更困难，我也不会感到惊讶。发展最早的20个用户的最好方法可能就是使用特洛伊木马：你让人们使用一种他们需要的应用程序，这个程序碰巧就是用某种新语言开发的。

外部因素

我们得先承认，确实有一个外部因素会影响到语言的流行。一种语言必须是某一个流行的计算机系统的脚本语言（scripting language），才会变得流行。Fortran和Cobol是早期IBM大型机的脚本语言。C是Unix的脚本语言，后来的Perl和Python也是如此。Tcl是Tk的脚本语言，Visual Basic是Windows的脚本语言，（某种形式的）Lisp是Emacs的脚本语言，PHP是网络服务器的脚本语言，Java和JavaScript是浏览器的脚本语言。

编程语言不是存在于真空之中。“编程”其实是及物动词，黑客一般都是为某个系统编程，在现实中，编程语言总是与它们依附的系统联系在一起的。所以，如果你想设计一种流行的编程语言，就不能只是单纯地设计语言本身，还必须为它找到一个依附的系统，而这个系统也必须流行。除非你只想用自己设计的语言取代那个系统现有的脚本语言。

这种情况导致的一个结果就是，无法以一种语言本身的优缺点评判这种语言。另一个结果则是，只有当一种语言是某个系统的脚本语言时，它才能真正成为编程语言。如果你对此很吃惊，觉得不公平，那么我会跟你说不必

大惊小怪。这就好比大家都认为，如果一种编程语言只有语法规则，没有一个好的实现（implementation），那么它就不能算完整的编程语言。这些都是很正常很合理的事情，编程语言本来就该如何。

当然，编程语言本来就需要一个好的实现，而且这个实现必须是免费的。商业公司愿意出钱购买软件，但是黑客作为个人不会愿意这样做，而你想让一种语言成功，恰恰就是需要吸引黑客。

编程语言还需要有一本介绍它的书。这本书应该不厚，文笔流畅，而且包含大量优秀的范例。布赖恩·柯尼汉和丹尼斯·里奇合写的《C程序设计语言》（*C Programming Language*）就是这方面的典范。眼下，我大概还能再加一句，这一类书籍之中必须有一本由O'Reilly公司出版发行。这正在变成是否能吸引黑客的前提条件了。

编程语言还应该要有在线文档。事实上，在线文档可以当做一本书来写，但是目前它无法取代实体书。实体书并没有过时，它们读起来很方便，而且出版社对书籍内容的审核是一种很有用的质量保证机制（虽然做得很不完美）。书店则是程序员发现和学习新语言的最重要的场所之一。

简洁

假定你的语言已经能够满足上面三项条件——一种免费的实现，一本相关书籍，以及语言所依附的计算机系统——那么还需要做什么才能使得黑客喜欢上你的语言？

黑客欣赏的一个特点就是简洁。黑客都是懒人，他们同数学家和现代主义建筑师一样，痛恨任何冗余的东西或事情。有一个笑话说，黑客动手写程序之前，至少会在心里盘算一下哪种语言的打字工作量最小，然后就选择使用该语言。这个笑话其实与真实情况相差无几。就算这真的是个笑话，语言的设计者也必须把它当真，按照它的要求设计语言。

简洁性最重要的方面就是要使得语言更抽象。为了达到这一点，首先你设计的必须是高级语言，然后把它设计得越抽象越好。语言设计者应该总是看着代码，问自己能不能使用更少的语法单位把它表达出来。如果你有办法让



Paul Graham

硅谷创业之父，经历颇为传奇。他大学在康奈尔大学主修哲学，研究生在哈佛大学主攻人工智能，博士期间又去哈佛艺术系旁听。拿到博士学位后，先后进入罗德岛设计学院和佛罗伦萨美术学院学习绘画，梦想成为画家。1995年，他与史上第一个蠕虫病毒“莫里斯蠕虫”的制作者Robert Morris开发出世界上第一个通过互联网使用的软件——Viaweb，该软件后来被雅虎以4900万美元兼并，并改名为Yahoo Store。结束在雅虎的任职后，他将自己对于技术和创业的观点写成系列文章，由O'Reilly集结出版，取名为《黑客与画家》。2005年，他与Morris等人合伙成立了Y Combinator（简称YC），它既是一个创业公司的孵化器，也是一个教导员，还是一个与投资人联系的中介。到目前为止，从YC“毕业”的创业公司共有200多家，已经失败的公司不到20%，远低于90%的业内平均水平。这些YC学员成为新一代硅谷创业公司的主流，他们组成了一张不断壮大的关系网，有人称其为“YC匪帮”。

许多不同的程序都能更简短地表达出来，那么这很可能意味着你发现了一种很有用的新抽象方法。

不要觉得为用户着想就是让他们使用像英语一样又长又啰唆的语法。这是不正确的做法，Cobol就是因为这个毛病而声名狼藉。如果你让黑客像下面这样求和：

```
add x to y giving z
而不是写成：
z=x+y
```

那么你就是在侮辱黑客的智商，或者自己作孽了。

简洁性是静态类型语言的力所不及之处。不考虑其他因素时，没人愿意在程序的头部写上一大堆的声明语句。只要计算机可以自己推断出来的事情，都应该让计算机自己去推断。举例来说，hello-world本应该是一个很简单的程序，但是在Java语言中却要写上一大堆东西，这本身就差不多可以说明Java语言设计得有问题了。^[1]

编程语言不是存在于真空之中。“编程”其实是及物动词，黑客一般都是为某个系统编程，在现实中，编程语言总是与他们依附的系统联系在一起。

单个的语法单位也应该很简短。Perl和Common Lisp在这方面是两个不同的极端。Perl的语法单位很短，导致它的代码可以拥挤得让人无法理解，而Common Lisp内置运算符的名称则长得可笑。Common Lisp的设计者们可能觉得文本编辑器会帮助用户自动填写运算符的长名称。但是这样做的代价不仅是增加了打字的工作量，还包括提高了阅读代码的难度，以及占用了更多的显示器空间。

可编程性（Hackability）

对黑客来说，选择编程语言的时候，还有一个因素比简洁更重要，那就是这种语言必须能够帮助自己做到想做的事。在编程语言的历史上，防止程序员做出“错误”举动的措施多得惊人。这是语言设计者很自以为是的危险举动，他们怎么知道程序员该做什么不该做什么？我认为，语言设计者应该假定他们的目标

用户是一个天才，会做出各种他们无法预知的举动，而不是假定目标用户是一个笨手笨脚的傻瓜，需要别人的保护才不会伤到自己。如果用户真的是傻瓜，不管你怎么保护他，他还是会搬起石头砸自己的脚。你也许能够阻止他引用另一个模块中的变量，但是你没法防止他日日夜夜不知疲倦地写出结构混乱的程序去解决完全错误的问题。

优秀程序员经常想做一些既危险又令人恼火的事情。所谓“令人恼火”，我指的是他们会突破设计者提供给用户的外部语义层，试着控制某些高级抽象的语言内部接口。比如，黑客喜欢破解，而破解就意味着深入内部，揣测原始设计者的意图。

你应该敞开心怀，欢迎这种揣测。对于制造工具的人来说，总是会有用户以违背你本意的方式使用你的工具。如果你制造的是编程语言这样高度组合的系统，那就更是如此了。许多黑客会用你做梦也想不到的方式改动你的语法模型。我的建议就是，让他们这样干吧，而且应该为他们创造便利，尽可能多地把语言的内部暴露在他们面前。

其实，黑客并不会彻底颠覆你的工具，在一个大型程序中，他可能只是对语言改造一两地方。但是，改动多少地方并不重要，重要的是他能够对语言进行改动。这可能不仅有助于解决一些特殊的问题，还会让黑客觉得很好玩。黑客改造语言的乐趣就好比外科医生摆弄病人内脏的乐趣，或者青少年喜欢用手挤破青春痘的那种感觉。至少对男生来说，某些类型的破坏非常刺激。针对青年男性读者的Maxim杂志每年出版一本特辑，里面一半是美女照片，另一半是各种严重事故的现场照片。这本杂志非常清楚它的读者想看什么。

一种真正优秀的编程语言应该既整洁又混乱。“整洁”的意思是设计得很清楚，内核由数量不多的运算符构成，这些运算符易于理解，每一个都有很完整的独立用途。“混乱”的意思是它允许黑客以自己的方式使用。C语言就是这样的例子，早期的Lisp语言也是如此。真正的黑客语言总是稍微带一点放纵不羁、不服管教的个性。

优秀的编程语言所具备的功能，应该会使得言必称“软件工程”的人感到非常不满、

频频摇头。与黑客语言形成鲜明对照的就是像Pascal那样的语言，它是井然有序的模范，非常适合教学，但是除此之外就没有很大用处了。

一次性程序

为了吸引黑客，一种编程语言必须善于完成黑客想要完成的各种任务。这意味着它必须很适合开发一次性程序。这一点可能出乎很多人的意料。

所谓一次性程序，就是指为了完成某些很简单的临时性任务而在很短时间内容写出来的程序。比如，自动完成某些系统管理任务的程序，或者（为了某项模拟任务）自动生成测试数据的程序，以及在不同格式之间转化数据的程序等。令人吃惊的是，一次性程序往往不是真的只用一次，就像二战期间很多美国大学造的一大批临时建筑后来都成了永久建筑。许多一次性程序后来也都变成了正式的程序，具备了正式的功能和外部用户。

我有一种预感，最优秀的那些大型程序就是这样发展起来的，而不是像胡佛水坝那样从一开始就作为大型工程来设计。一下子从无到有做出一个大项目是一件很恐怖的事。当人们接手一个巨型项目时，很容易被它搞得一蹶不振。最后，要么是项目陷入僵局，要么是做出一个规模小、性能差的东西。你想造一片闹市，却只做出一家商场；你想建一个罗马，却只造出一个巴西利亚；你想发明C语言，却只开发出Ada。

开发大型程序的另一个方法就是从一次性程序开始，然后不断地改进。这种方法比较不会让人望而生畏，程序在不断的开发之中逐渐进步。一般来说，使用这种方法开发程序，一开始用什么编程语言，就会一直用到最后，因为除非有外部政治因素的干预，程序员很少会中途更换编程语言。所以，我们就有了一个看似矛盾的结论：如果你想设计一种适合开发大型项目的编程语言，就必须使得这种语言也适合开发一次性程序，因为大型项目就是从一次性程序演变而来的。

Perl就是一个鲜明的例子。它不仅仅设计成适合开发一次性程序，而且它本身就很像一次性程序。最初的Perl只是好几个生成表格的工具收集在一起而已。后来程序员用它写一次性程序，当那些程序逐渐发展壮大后，Perl才随之发

展成了一种正式的编程语言。到了Perl 5，这种语言才适合开发重要的程序，但是在此之前它已经广为流行了。

什么样的语言适合写一次性程序？首先，它必须很容易装备。一次性程序是你只想在一小时内写出来的程序，所以它不应该耗费很多时间安装和配置，最好已经安装在你的电脑上了。它必须是想用就用的。C语言可以想用就用，因为它是操作系统的一部分；Perl可以想用就用，因为它本来就是一种系统管理工具，操作系统已经默认安装它了。

很容易装备不仅仅指很容易安装或者已经安装，还指很容易与使用者互动。一种有命令行界面、可以实时反馈的语言就具有互动性，那些必须先编译后使用的语言就不具备互动性。受欢迎的编程语言应该是前者，具有良好的互动性，可以快速得到运行结果。

一次性程序的另一个特点就是简洁。对黑客来说，这一点永远有吸引力。如果考虑到你最多只打算在这个程序上耗费一个小时，这一点就更重要了。P

注：

[1] hello-world程序的唯一作用就是显示出“Hello, world!”这句话。使用Java语言，你需要这样写：

```
public class Hello
{
    public static void main(String[] args)
    {
        System.out.println("Hello, world!");
    }
}
```

如果你从来没有接触过编程，看到上面的代码可能会很奇怪，让计算机显示一句话为什么要搞得这么复杂？有意思的是，资深程序员的反应与你一样。



本文节选自Paul Graham的文集《黑客与画家》，主要介绍黑客及优秀程序员的爱好和动机，讨论黑客成长、黑客对世界的贡献以及编程语言和黑客工作方法等话题。中文版将由人民邮电出版社北京图灵文化发展有限公司出版，特此感谢图灵公司授权。

RCFile——用于Facebook数据仓库的高效存储结构

文 / 杨栋

本文介绍了Facebook公司数据分析系统中的RCFile存储结构，该结构集行存储和列存储的优点于一身，在MapReduce环境下的大规模数据分析中扮演重要角色。

Facebook曾在2010 ICDE (IEEE International Conference on Data Engineering) 会议上介绍了数据仓库Hive。Hive存储海量数据在Hadoop系统中，提供了一套类数据库的数据存储和处理机制。它采用类SQL语言对数据进行自动化管理和处理，经过语句解析和转换，最终生成基于Hadoop的MapReduce任务，通过执行这些任务完成数据处理。图1显示了Hive数据仓库的系统结构。

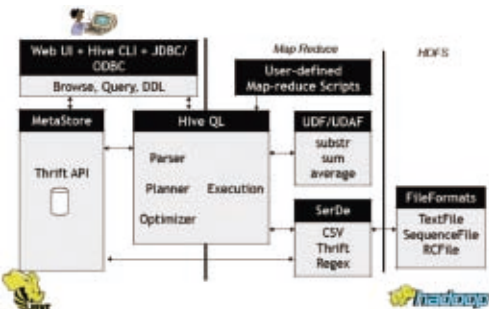


图1 Hive数据仓库的系统结构

基于MapReduce的数据仓库在超大规模数据分析中扮演了重要角色，对于典型的Web服务供应商，这些分析有助于它们快速理解动态的用户行为及变化的用户需求。

数据存储结构是影响数据仓库性能的关键因素之一。Hadoop系统中常用的文件存储格式有支持文本的TextFile和支持二进制的SequenceFile等，它们都属于行存储方式。Facebook工程师发表的RCFile: A Fast and Space-efficient Data Placement Structure in MapReduce-based Warehouse Systems一文，介绍了一种高效的数据存储结构——RCFile (Record Columnar File)，并将其应用于Facebook的数据仓库

Hive中。与传统数据库的数据存储结构相比，RCFile更有效地满足了基于MapReduce的数据仓库的四个关键需求，即Fast data loading、Fast query processing、Highly efficient storage space utilization和Strong adaptivity to highly dynamic workload patterns。

数据仓库的需求

基于Facebook系统特征和用户数据的分析，在MapReduce计算环境下，数据仓库对于数据存储结构有四个关键需求。

Fast data loading

对于Facebook的产品数据仓库而言，快速加载数据（写数据）是非常关键的。每天大约有超过20TB的数据上传到Facebook的数据仓库，由于数据加载期间网络和磁盘流量会干扰正常的查询执行，因此缩短数据加载时间是非常必要的。

Fast query processing

为了满足实时性的网站请求和支持高并发用户提交查询的大量读负载，查询响应时间是非常关键的，这要求底层存储结构能够随着查询数量的增加而保持高速的查询处理。

Highly efficient storage space utilization

高速增长的用户活动总是需要可扩展的存储容量和计算能力，有限的磁盘空间需要合理管理海量数据的存储。实际上，该问题的解决方案就是最大化磁盘空间利用率。

Strong adaptivity to highly dynamic workload patterns

同一份数据集会供给不同应用的用户，通过各种方式来分析。某些数据分析是例行过程，按照某种固定模式周期性执行；而另一些则是从中间平台发起的查询。大多数负载不遵循任何规则模式，这需要底层系统在存储空间有限的前提下，对数据处理中不可预知的动态数据具备高度的适应性，而不是专注于某种特殊的负载模式。

MapReduce存储策略

要想设计并实现一种基于MapReduce数据仓库的高效数据存储结构，关键挑战是在MapReduce计算环境中满足上述四个需求。在传统数据库系统中，三种数据存储结构被广泛研究，分别是行存储结构、列存储结构和PAX混合存储结构。

上面这三种结构都有其自身特点，不过简单移植这些数据库导向的存储结构到基于MapReduce的数据仓库系统并不能很好地满足所有需求。

行存储

如图2所示，基于Hadoop系统行存储结构的优点在于快速数据加载和动态负载的高适应能力，这是因为行存储保证了相同记录的所有域都在同一个集群节点，即同一个HDFS块。不过，行存储的缺点也是显而易见的，例如它不能支持快速查询处理，因为当查询仅仅针对多列表中的少数几列时，它不能跳过不必要的列读取；此外，由于混合着不同数据值的列，行存储不易获得一个极高的压缩比，即空间利用率不易大幅提高。尽管通过熵编码和利用列相关性能够获得一个较好的压缩比，但是复杂数据存储实现会导致解压开销增大。

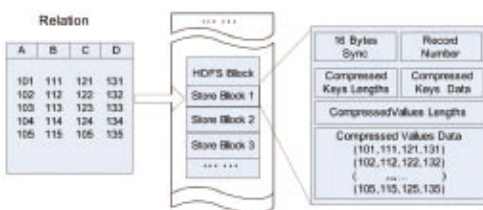


图2 HDFS块内行存储的例子

列存储

图3显示了在HDFS上按照列组存储表格的例子。在这个例子中，列A和列B存储在同一列组，而列C和列D分别存储在单独的列组。查询时列存储能够避免读不必要的列，并且压缩一个列中的相似数据能够达到较高的压缩比。然而，由于元组重构的较高开销，它并不能提供基于Hadoop系统的快速查询处理。列存储不能保证同一记录的所有域都存储在同一集群节点，例如图2的例子中，记录的4个域存储在位于不同节点的3个HDFS块中。因此，记录的重构将导致通过集群节点网络的大量数据传输。尽管预先分组后，多个列在一起能够减少开销，但是对于高度动态的负载模式，它并不具备很好的适应性。除非所有列组根据可能的查询预先创建，否则对于一个查询需要一个不可预知的列组合，一个记录的重构或许需要2个或多个列组。再者由于多个组之间的列交叠，列组可能会创建多余的列数据存储，这导致存储利用率的降低。

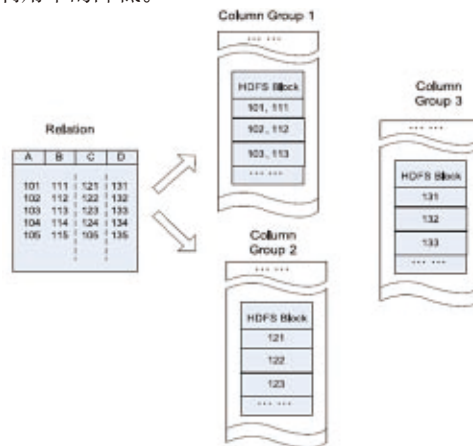


图3 HDFS块内列存储的例子

PAX混合存储

PAX存储模型（用于Data Morphing存储技术）使用混合存储方式，目的在于提升CPU Cache性能。对于记录中来自不同列的多个域，PAX将它们放在一个磁盘页中。在每个磁盘页中，PAX使用一个迷你页来存储属于每个列的所有域，并使用一个页头来存储迷你页的指针。类似于行存储，PAX对多种动态查询有很强的适应能力。然而，它并不能满足大型分布式系统对于高存储空间利用率和快速查询处理的需求，原因在于：首先，PAX没有数据压缩

的相关工作，这部分与Cache优化关系不大，但对于大规模数据处理系统是非常关键的，它提供了列维度数据压缩的可能性；其次，PAX不能提升I/O性能，因为它不能改变实际的页内容，该限制使得大规模数据扫描时不易实现快速查询处理；再次，PAX用固定的页作为数据组织的基本单位，按照这个大小，在海量数据处理系统中，PAX将不会有效存储不同大小类型的数据域。

本文介绍的是RCFile数据存储结构在Hadoop系统上的实现。该结构强调：第一，RCFile存储的表是水平划分的，分为多个行组，每个行组再被垂直划分，以便每列单独存储；第二，RCFile在每个行组中利用一个列维度的数据压缩，并提供一种Lazy解压（decompression）技术来在查询执行时避免不必要的列解压；第三，RCFile支持弹性的行组大小，行组大小需要权衡数据压缩性能和查询性能两方面。

RCFile的设计与实现

RCFile（Record Columnar File）存储结构遵循的是“先水平划分，再垂直划分”的设计理念，这个想法来源于PAX。它结合了行存储和列存储的优点：首先，RCFile保证同一行的数据位于同一节点，因此元组重构的开销很低；其次，像列存储一样，RCFile能够利用列维度的数据压缩，并且能跳过不必要的列读取。图4是一个HDFS块内RCFile方式存储的例子。

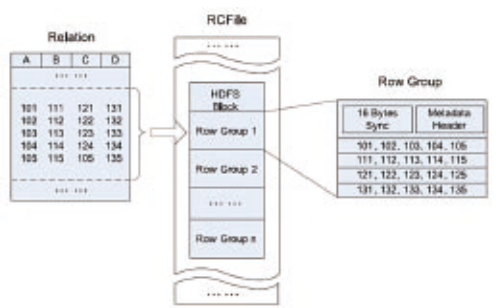


图4 HDFS块内RCFile方式存储的例子

数据格式

RCFile在HDFS分布式文件系统之上设计并实现，如图4所示，RCFile按照下面的数据格式来存储一张表。

- RCFile基于HDFS架构，表格占用多个HDFS块。

- 每个HDFS块中，RCFile以行组为基本单位来组织记录。也就是说，存储在一个HDFS块中的所有记录被划分为多个行组。对于一张表，所有行组大小都相同。一个HDFS块会有一个或多个行组。

- 一个行组包括三个部分。第一部分是行组头部的同步标识，主要用于分隔HDFS块中的两个连续行组；第二部分是行组的元数据头部，用于存储行组单元的信息，包括行组中的记录数、每个列的字节数、列中每个域的字节数；第三部分是表格数据段，即实际的列存储数据。在该部分中，同一列的所有域顺序存储。从图4可以看出，首先存储了列A的所有域，然后存储列B的所有域等。

压缩方式

RCFile的每个行组中，元数据头部和表格数据段分别进行压缩。

- 对于所有元数据头部，RCFile使用RLE（Run Length Encoding）算法来压缩数据。由于同一列中所有域的长度值都顺序存储在该部分，RLE算法能够找到重复值的长序列，尤其对于固定的域长度。

- 表格数据段不会作为整个单元来压缩，相反每个列被独立压缩，使用Gzip压缩算法。RCFile使用重量级的Gzip压缩算法，是为了获得较好的压缩比，而不使用RLE算法的原因在于此时列数据非排序。此外，由于Lazy压缩策略，当处理一个行组时，RCFile不需要解压所有列。因此，相对较高的Gzip解压开销可以减少。

尽管RCFile对表格数据的所有列使用同样的压缩算法，不过如果使用不同的算法来压缩不同列或许效果会更好。RCFile将来的工作之一可能就是根据每列的数据类型和数据分布来自适应选择最好的压缩算法。

数据追加

RCFile不支持任意方式的数据写操作，仅提供一种追加接口，这是因为底层的HDFS当前仅仅支持数据追加写文件尾部。数据追加方法描述如下。

- RCFile为每列创建并维护一个内存column holder，当记录追加时，所有域被分发，每个域追加到其对应的column holder。此外，RCFile在元数据头部中记录每个域对应的元数据。

- RCFile提供两个参数来控制在刷写到磁盘之前，内存中缓存多少个记录。一个参数是记录数的限制，另一个是内存缓存的大小限制。

- RCFile首先压缩元数据头部并写到磁盘，然后分别压缩每个column holder，并将压缩后的column holder刷写到底层文件系统中的行组中。

数据读取和Lazy解压

在MapReduce框架中，mapper将顺序处理HDFS块中的每个行组。当处理一个行组时，RCFile无需全部读取行组的全部内容到内存。相反，它仅仅读元数据头部和给定查询需要的列。因此，它可以跳过不必要的列以获得列存储的I/O优势。例如，表tbl(c1, c2, c3, c4)有4个列，做一次查询“SELECT c1 FROM tbl WHERE c4 = 1”，对每个行组，RCFile仅仅读取c1和c4列的内容。在元数据头部和需要的列数据加载到内存中后，它们需要解压。元数据头部总会解压并在内存中维护直到RCFile处理下一个行组。然而，RCFile不会解压所有加载的列，相反，它使用一种Lazy解压技术。

Lazy解压意味着列将不会在内存解压，直到RCFile决定列中数据真正对查询执行有用。由于查询使用各种WHERE条件，Lazy解压非常有用。如果一个WHERE条件不能被行组中的所有记录满足，那么RCFile将不会解压WHERE条件中不满足的列。例如，在上述查询中，所有行组中的列c4都解压了。然而，对于一个行组，如果列c4中没有值为1的域，那么就无需解压列c1。

行组大小

I/O性能是RCFile关注的重点，因此RCFile需要行组够大并且大小可变。行组大小和下面几个因素相关。

- 行组大的话，数据压缩效率会比行组小时更有效。根据对Facebook日常应用的观察，当行组大小达到一个阈值后，增加行组大小并不能进一步增加Gzip算法下的压缩比。

- 行组变大能够提升数据压缩效率并减少存储量。因此，如果对缩减存储空间方面有强烈需求，则不建议选择使用小行组。需要注意的是，当行组的大小超过4MB，数据的压缩比将趋于一致。

- 尽管行组变大有助于减少表格的存储规模，但是可能会损害数据的读性能，因为这样减少了Lazy解压带来的性能提升。而且行组变大会占用更多的内存，这会影响并发执行的其他MapReduce作业。

考虑到存储空间和查询效率两个方面，Facebook选择4MB作为默认的行组大小，当然也允许用户自行选择参数进行配置。

小结

本文简单介绍了RCFile存储结构，其广泛应用于Facebook公司的数据分析系统Hive中。首先，RCFile具备相当于行存储的数据加载速度和负载适应能力；其次，RCFile的读优化可以在扫描表格时避免不必要的列读取，测试显示在多数情况下，它比其他结构拥有更好的性能；再次，RCFile使用列维度的压缩，因此能够有效提升存储空间利用率。

为了提高存储空间利用率，Facebook各产品线应用产生的数据从2010年起均采用RCFile结构存储，按行存储（SequenceFile/TextFile）结构保存的数据集也转存为RCFile格式。此外，Yahoo公司也在Pig数据分析系统中集成了RCFile，RCFile正在用于另一个基于Hadoop的数据管理系统Howl（<http://wiki.apache.org/pig/Howl>）。而且，根据Hive开发社区的交流，RCFile也成功整合加入其他基于MapReduce的数据分析平台。

有理由相信，作为数据存储标准的RCFile，将继续在MapReduce环境下的大规模数据分析中扮演重要角色。P



杨栋

百度分布式研发工程师，主要从事分布式计算系统和分布式表格系统的开发和研究。

责任编辑：董世晓（dongsx@csdn.net）

浅析.NET应用程序的内存问题

文 / 许大伟

曾经困扰很多开发人员的“内存泄漏”问题目前仍值得我们关注，本文介绍了一些在.NET世界里的典型内存问题及其解决方案。

性能对于一个企业级产品的重要性不言而喻，因为它直接关系到产品是否稳定、在线人数是否满足客户需求、产品是否存在性能瓶颈等问题，所以性能测试已经逐渐成为产品发布前必不可少的一个步骤。于是乎，性能调优及性能问题的排查也就成为开发人员工作的重要一环。

衡量性能的一个重要指标是内存使用是否过大和是否平稳，因为如果内存没有被及时释放的话，可能会马上或者即将抛出OutOfMemory的错误。即使现在的服务器有较大的内存，内存的大量占用也会导致整个系统吞吐率下降，这样的程序质量势必会在客户或用户心中打上一个大大的问号。

最近，我所参与的一个基于Sharepoint的项目终于通过了性能测试，顺利发布，其间也是经历了种种坎坷。从最开始在某些环境下几个QA简单的人工测试就导致系统运行减慢，到开始使用LoadRunner模拟50个、300个用户测出内存溢出，再到最后终于通过2000个并发用户测试。我们逐渐深刻地认识到，即使是使用托管语言做开发，“内存泄漏”这个曾经很困扰许多非托管语言的开发人员的问题，也依然值得我们去注意。

本文作为对项目性能调优经验的总结，将描述一些发生在.NET世界里的典型内存问题。由于.NET Framework许多细节尚未有公开资料，我也主要是基于互联网上许多文章及自己工作中的一些经验总结而来，若文章给您的工作造成不便，敬请谅解。另外，本文中的许多测试均是基于.NET Framework 2.0，所以不排除在其他版本中有细微变化。

背景知识

本质上，.NET应用程序的内存使用主要就三个步骤：通过VirtualAlloc函数执行预留（Reserving）和提交（Committing），以及通过VirtualFree函数来进行释放（Freeing）。因此有以下两种情况可以导致.NET应用程序抛出OOM错误：一种是CLR发现没有足够的虚拟内存可供使用（这个限定值在32位操作系统为64MB，64位为1GB），这就是为什么有时我们发现系统还有足够的物理内存，却依然收到OOM错误的原因；另外一种可能就是没有足够的物理内存^[1]。

在.NET程序的运行过程中，贯穿始终的重要机制就是GC（Garbage Collection，垃圾收集），所以我们简单回顾一下它。GC是CLR采用了一些算法进行清理托管内存的过程，可以简单理解如下：

1. 挂起Execution Engine；
2. 标记所有的不可达对象（Unreachable Object）；
3. 清除所有被标记的对象；
4. 移动堆栈顶部未被删除的Unpinned对象到堆栈底部；
5. 重新启动Execution Engine。

GC在多数情况下工作良好，使得部分开发人员认为GC已经帮助我们解决了一切问题，不必考虑代码在内存占用方面的事情。岂不知这很有可能已经在你的代码中埋了颗定时炸弹，一旦系统负载加上了来，将会触发一些性能问题。

为了叙述清楚，我们先给出一个简单的.NET应用程序的内存逻辑示意图（不考虑共享

内存)，如图1所示。

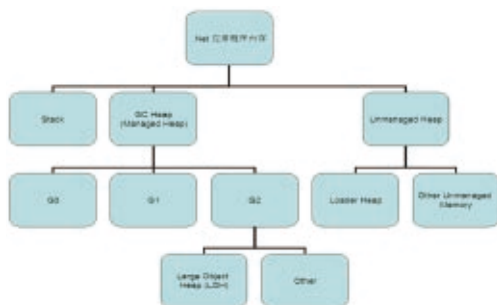


图1 .NET应用程序的内存逻辑示意图

对于内存泄漏，首要的事情是确定哪一部分的内存发生了泄漏，这个可以通过Perfmon^[2]看出一些端倪，通常主要通过以下两个计数器来进行推断。

- Process/Private Bytes，程序所独占的不能共享的内存，可以简单地理解为图1中的根节点“ .NET应用程序内存”。

- NET CLR Memory / # Bytes in All Heaps，对应于图1中的“GC Heap”。

如果“Private Bytes”持续增长，而“Bytes in All Heaps”保持平稳，则说明非托管部分内存发生泄漏，如图2所示。



图2 非托管部分内存发生泄漏

如果“Private Bytes”持续增长，而“Bytes in All Heaps”也持续增长，且两者增长比率近似，则说明托管部分内存发生泄漏，如图3所示，需要注意的是，GC只会在执行GC后才更新它的相关计数器，所以在图3中我们可以看到“#Bytes in all Heaps”中间会有平稳的时间段。

如果“Private Bytes”持续增长，而“Bytes in All Heaps”也持续增长，且两者增长比率明显不同，则很有可能是托管内存和非托管内存同时发生了泄漏。如图4所示。

需要补充的是，我们平常通过“Windows

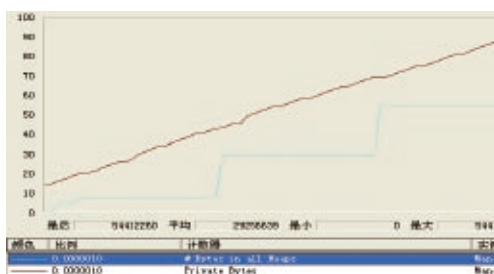


图3 托管部分内存发生泄漏

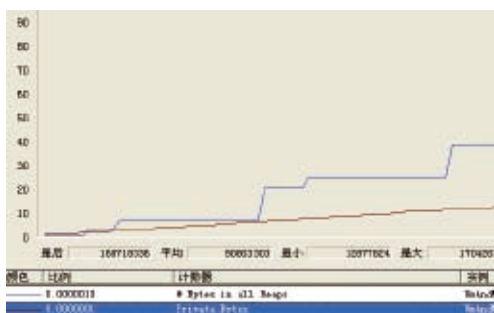


图4 托管内存和非托管内存同时发生了泄漏

任务管理器（TaskManager）”可以看到“内存使用（Mem Usage）”和“虚拟内存大小（Virtual Memory Size）”两列，其名称与Perfmon的一些计数器过于相像，容易引起混淆，我们在此做简单澄清。任务管理器中的“内存使用”代表的是进程的“Working Set”，仅代表当前程序可以不通过Paging就能直接访问的物理内存^[3]，其中也包含了共享内存，值与计数器“Process/Working Set”相当；“虚拟内存大小”则与计数器“Process/Private Bytes”一样，代表分配给该进程的私有的虚拟内存使用大小（Committed Visual Memory）；而计数器“Process/Virtual Bytes”则表示该进程在虚拟内存上预留和提交的地址空间。

虽然堆栈也会由于编程导致的错误致使递归调用的死循环从而堆栈溢出，或者线程死锁致使线程堆栈一直被保留，但这些问题一般容易在开发阶段或者早期测试阶段发生，故本文不多做介绍。我们主要关注发生在非托管堆和托管堆区域的内存泄漏。

非托管堆

非托管堆主要包含非托管代码所分配、占用的内存以及托管代码运行所用到的类型系统相关信息（IL Code、JITed Code、Method Tables……），此处内存也不受GC维护，一般容易产

生以下几个问题。

考虑有这么一个托管对象，其所占托管内存非常小，但又占着较大的非托管内存，这样即使该对象已正确实现了Finalize方法，也仍然会有内存问题。对于此种对象，CLR在GC时只考虑托管内存，从而未能认识到GC的必要性，导致内存不断上涨，引起系统性能下降甚至OOM。典型的如FCL中的Bitmap、SharePoint编程对象中的SPSite和SPWeb^[4]，解决方法是在访问完实现了IDisposable接口的对象后，要及时地调用Dispose方法释放它们。另外，我们在代码中实现此类对象时，也可以调用微软在.NET 2.0中新增加的方法public static void AddMemoryPressure (Long bytesAllocated)来避免这个问题。

实际上这也正是许多最佳实践推荐实现Finalize方法的同时也实现IDisposable接口的原因，也是即使该对象已经实现了Finalize方法，我们却仍然需要保证Dispose方法被及时调用的原因。据此，我们不妨强制要求代码中一定要使用Using块来访问IDisposable对象。

Loader Heap内存泄漏。Loader Heap主要是存储元数据相关信息，且它所占内存只能在AppDomain被卸载时释放，通常我们的应用程序所依赖的程序集是有限的，所有此处的内存问题主要是由动态程序集（Dynamic Assembly）导致的。典型问题如XmlSerializer对象，参考MSDN的描述：

To increase performance, the XML serialization infrastructure dynamically generates assemblies to serialize and deserialize specified types. The infrastructure finds and reuses those assemblies. This behavior occurs only when using the following constructors: System.Xml.Serialization.XmlSerializer(Type) System.Xml.Serialization.XmlSerializer(Type, String)

其内部在序列化/反序列化时动态生成程序集，如果没有按照上面的说明构造该对象，同一个类型的对象多次XML序列化时会生成多份临时程序集，而由于它们只能在AppDomain被卸载时才能被释放，从而导致内存不断被占用，得不到释放。

同样的问题也存在于WCF的代码之中，每次代码生成Proxy实例来访问Service时，其内部也会生成临时的动态程序集，最终可以导致OOM发生。该问题已经在.NET Framework 3.0Sp1中被解决，当然我们也可以通过自己缓

存Proxy实例来解决该问题。

进一步引申思考，可以联想到其他使用了动态程序集的技术，如XSLT、CodeDom等，应该避免使用或者小心使用它们。对于CodeDom，我们可以使用微软在.NET Framework 2.0中新增加的Reflection.Emit来替代，它可以避免生成动态程序集。对于动态程序集泄漏还有一个比较麻烦的解决方法，就是动态创建一个AppDomain，然后将动态程序集加载到它里面，最后不用时可以通过卸载该AppDomain来释放内存（印象中好像Nunit就使用了这个方法）。

可以通过Perfmon中的计数器“.NET CLR Loading\Current Assemblies”来发现该问题。

Finalize没有被正常执行。我们知道通常Finalize方法是用来释放非托管资源的，对于Finalize方法必须加以额外的注意，这是由于它有以下特性。

- 它是被CLR一个^[5]特殊的高优先级的线程专门调用的。
- 不同对象的Finalize方法的调用顺序不被保证。
- 实现了Finalize方法的对象至少会被提升到G1后才有可能被释放，该对象所引用的其他对象的生命周期也会同时被延长。

基于以上认识，我们会发现，如果有任何一个Finalize方法由于编程不当被堵塞（例如在Finalize方法中不断地尝试写日志到一个已经宕掉的数据库中，或者错误地调用了Lock、Thread.Join()等），那么执行Finalize方法的线程也将被堵塞，导致其他Finalize方法不能被执行，继而造成相应的托管和非托管资源得不到释放。因此首先我们不建议使用Finalize方法，其次如果不得不实现Finalize方法，必须确保方法实现中不会阻塞Finalize调用线程。

该问题我们可以通过WinDBG打开内存显著上升前后的Snapshot，然后执行命令finalizequeue对比输出，如果发现内存上升后的Snapshot中终止化队列（Frecheable queue）中对象比之前有大量增长，则说明Finalize线程被阻塞，导致Finalize对象不能被最终释放。

托管堆

尽管GC Heap受GC的维护，但不当的代码仍然可以造成它的内存泄漏。

LOH产生内存碎片，最终导致内存泄漏。我们知道LOH是用来存储大于85KB的大对象。对LOH部分的内存进行GC的过程基本与背景知识部分介绍的GC过程相同，唯一的区别是清理LOH的对象后，CLR并不会移动、压缩此处内存（这是由于在内存中移动大对象较损耗性能），这样在LOH中就容易产生内存碎片，如图5所示。在继续分配大对象时，由于内存碎片的存在，致使已经Free的内存不能得到有效利用，使得系统看起来仍然有足够的空闲内存，但程序依然抛出内存溢出错误。



图5 LOH中产生的内存碎片

以上所述场景在现实中并不少见，目前该问题已被提交给微软^[6]，在未来.NET Framework版本中可能会修改或改善该问题。在当前的编码中，只能是尽量避免使用大对象，例如准备存储100KB的数据的话，可以将该数据分成10个对象来存储，每个对象仅存储10KB数据。

此问题可以通过计数器“.NET CLR Memory/Large Object Size”来进行确认。

需要注意的是除了LOH中的对象不会在GC过程中被压缩移动外，SOH（Small Object Heap）的Pinned对象也有相同的特性。例如在unsafe代码中fixed(int *pInt = &objA.X)的作用域内，objA所引用的对象将被保证不能移动。那么如果在该作用域内执行一次垃圾回收、内存分配，也可能导致内存碎片。

未释放的事件引用。不及时释放引用而导致内存泄漏是一个较常见且很隐蔽的问题。假设在

```
objA.Event += new EventHandler(objB.EventProcMethod)
```

代码里创建了对象A和对象B，且有以下代码：

如果objB对象已经不再被使用，而objA仍然被使用，则由于事件引用的存在，导致objB对象也得不到释放。这种问题较容易发生在Web开发中，objA对象可能存储在Session、Cache或某个Static对象中，也可能为Cache对象的回调方法所在实例。解决方法一是及时地使用“-=”释放事件引用，二是创建EventHandler实例时不要使用实例方法，而采用静态方法。

不当地使用Session。我们知道Session是用户级别的，这意味着Session中存储的内容如

果没有及时清除的话，随着不断有用户访问，Session所占内存会越来越大。由于Session没有针对某个键值的过期设置，只有一个全局的过期设置，默认情况下，可以在web.config中配置Session过期时间，但同时我们也可以通过代码来设置这个过期时间，因此容易造成一个问题：明明web.config配置的是10分钟，却在代码又被开发人员错误的配置成3小时，导致所有的Session都不得不保存至少3小时。在此期间内，Session极有可能会被耗掉所有可用的内存，导致OOM发生。

通过以上分析，我们可以认识到.NET应用程序产生OOM的原因是各式各样的，但多数都是由于我们不小心地编码造成的。由此我认为，在产品发布前，进行性能测试是非常必要的，目前有许多工具可以提供类似的测试，商业软件有LoadRunner，开源工具有OpenSTA。另外在我们平时的开发工作中，也应在掌握相关理论知识后加强Code Review/Sharing，尽可能地将问题消灭在产品发布前。P

参考文献

- [1] 尽管在64位操作系统下理论上有足够的虚拟内存（64GB），但需要注意，如果虚拟内存使用太高，容易造成操作系统在虚拟内存和物理内存之间的频繁Paging，导致CPU占有率增高。
- [2] Perfmon, 微软提供的免费的系统性能监测工具，Windows XP/2003/2008等中均已默认包含，可以通过“开始→运行→Perfmon”打开。
- [3] 这点也可以通过观察在一个程序最小化后，其值变小来得到佐证。
- [4] 微软发布的SPDisposeCheck工具用于检查SharePoint对象有可能导致的内存溢出问题。
- [5] 在多处理器环境下，可以启用Server GC模式，其为每个处理器启用一个处理线程。Concurrent GC模式及其在.NET 4.0中的一些改进，可进一步参考其他文章。
- [6] 详情请参考<http://connect.microsoft.com/VisualStudio/feedback/details/521147/large-object-heap-fragmentation-causes-outofmemoryexception>。



许大伟

毕业于长安大学信息与计算科学专业，目前就职于奥博杰天软件（北京）有限公司西安研发中心，主要从事SharePoint与Documentum相关软件的开发。

责任编辑：董世晓（dongsx@csdn.net）

调整Netfilter内核模块以限制P2P连接

文 / 李晨光

通过调整Linux内核中Netfilter模块的方式，可以对P2P连接进行有目的的控制，从而为带宽资源有限的中小型企业用户提供了一个选择。

中小企业、办公室用户或者带宽资源有限的用户一般采用NAT（Network Address Translation，网络地址转换）来接入Internet。如果内部用户肆无忌惮地使用P2P下载或在线播放，势必会影响到其他用户正常使用网络。如果把P2P的端口全部封杀掉，用户就享受不到P2P所带来的好处。Linux由于其开放源代码的特性，近年来得到了迅速的发展，作为一个高性能的网络操作系统，Linux内核中的防火墙扮演着非常重要的角色。

Netfilter结构框架

最开始的ipfwadm是AlanCox在Linux Kernel发展的初期，从FreeBSD的内核代码中移植过来的。后来经历了ipchains，再经由Paul Russel在Linux Kernel 2.3系列的开发过程中发展了Netfilter这个架构。而用户空间的防火墙管理工具，也相应地发展为iptables。在经历了Linux Kernel2.4和2.6的发展之后，可以说，Netfilter iptabtes经受住了大量用户广泛使用的考验。本文将基于Linux 2.6的内核来进行叙述。

Netfilter是Linux2.6内核实现的防火墙框架，比以前任何一版Linux内核的防火墙子系统都要完善强大。Nefilter提供了一个抽象、通用化的框架，该框架定义的一个子功能实现的就是包过滤子系统Netfilter由一系列基于协议栈的钩子组成，这些钩子都对应某一具体的协议。

Netfilter在IPv4中的结构

尽管Linux 2.6支持对IPv4、IPv6及DECnet的钩子，但本文只提及IPv4的钩子。IPv4协议栈为了实现对Nefilter架构的支持，在IP包在

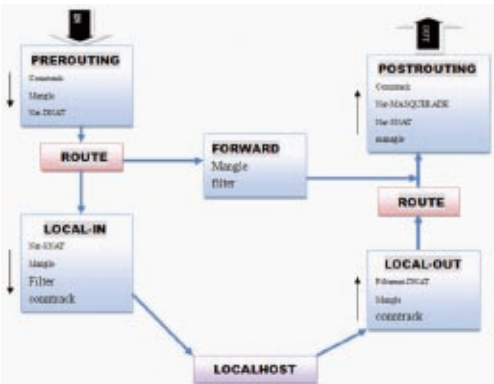


图1 Netfilter在IPv4中的结构

IPv4协议栈上的游历路线（如图1所示）中选择了五个检查点，可以在linux/netfilter_ipv4.h里面找到这些符号的定义，表1列出了这些钩子。

表1 IPv4中定义的钩子

钩子名称	调用时机
NF_IP_PRE_ROUTING	完整性校验之后，路由决策之前
NF_IP_LOCAL_IN	目的地为本机，路由决策之后
NF_IP_FORWARD	数据包要到达另外一个接口去
NF_IP_LOCAL_OUT	本地进程的数据，发送出去的过程中
NF_IP_POST_ROUTING	向外流出的数据上线之前

在这五个检查点上，各引入了一行对NF_HOOK()宏函数的一个相应的调用。如果没配置防火墙，NF_HOOK()便从Netfilter模块转回到IPv4协议栈继续往下处理；如果配置了防火墙，NF_HOOK()就转去调用nf_hook_slow()函数，该函数便会按顺序调用在该检查点注册的钩子函数，不管钩子函数对数据包做了哪些处理，都必须返回表2中的一个预定义好的返回值。NF_IP_PRE_ROUTING钩子是数据包接收后第一个调用的钩子程序，这个钩子在后面编写的模块当中将会被用到。

NF_DROP表示要丢弃这个数据包，并且为

表2 Netfilter的返回值

返回值	含义
NF_DROP	丢弃这个数据包
NF_ACCEPT	保留这个数据包
NF_STOLEN	忘掉这个数据包
NF_QUEUE	让这个数据包在用户空间排队
NF_REPEAT	再次调用这个钩子函数

这个数据包申请的所有资源都要得到释放。NF_ACCEPT告诉Netfilter到目前为止，这个数据包仍然可以被接受，应该将它移到网络堆栈的下一层。NF_STOLEN是非常有趣的一个返回值，它告诉Netfilter让其忘掉这个数据包。也就是说钩子函数会在这里对这个数据包进行完全的处理，而Netfilter就应该放弃任何对它的处理了。然而这并不意味着为该数据包申请的所有资源都要释放掉。这个数据包和它各自的sk_buff结构体依然有效，只是钩子函数从Netfilter夺取了对这个数据包的掌控权。最后一个返回值NF_REPEAT，就是当用户改变了该数据包包头的某些信息，那可以请求Netfilter再次调用这个钩子函数对它进行操作。

注册和注销Netfilter钩子函数

在上面提到了nf_hook_slow()函数会按顺序调用在该检查点注册的钩子函数，那钩子函数是怎样注册的呢？注册一个钩子函数是一个围绕nf_hook_ops结构体的很简单的过程，在linux/netfilter.h中有这个结构体的定义：

```
struct nf_hook_ops
{
    struct list_head list;
    nf_hookfn *hook;
    int pf;
    int hooknum;
    int priority;
};
```

这个结构体的成员列表主要是用来维护注册的钩子函数列表的，对于用户来说，在注册时并没有多么重要。list是一个有prev和next两个域的双向链表，各检查点的钩子函数就是通过它来按照priority的值由小到大链接在一起的，nf_hook_slow()函数就是靠遍历这个表来调用该检查点的钩子函数的。hook是指向nf_hookfn函数的指针，也就是为这个钩子将要调用的所有函数。nf_hookfn同样定义在linux/netfilter.h这个文件中。pf字段指定了协议簇（Protocol family），linux/socket.h中定义了可用的协议簇。对于IPv4只使用PF_INET。hooknum域指明了为哪个检查点安装这个函数，也就是表1中所列出的条目中的一个。priority域表示在运行时这个钩子函数执行的

顺序。我们选择NF_IP_PRI_FIRST这个最高优先级来运行编写的内核模块。

连线跟踪

每个网络连接包括以下信息：源地址、目的地址、源端口和目的端口、套接字对（Socket Pairs）协议类型、连接状态（TCP协议）和超时时间等。防火墙把这些信息叫做状态（Stateful），能够检测每个连接状态的防火墙叫做状态包过滤防火墙。它除了能够完成简单包过滤防火墙的包过滤工作外，还在自己的内存中维护一个跟踪连接状态的表，比简单包过滤防火墙具有更大的安全性。

以前，经常需要打开1024以上的所有端口来放行应答的数据。现在有了连线跟踪，就不需再这样了，可以只开放那些有应答数据的端口，其他的都可以关闭。这样安全多了。

连线跟踪工作原理

连线跟踪的工作原理并不十分复杂，以一个防火墙接收到一个初始化TCP连接的SYN包为例，这个带有SYN的网络协议包将被防火墙的规则库检查。

该数据包将在规则库中依次序进行比较。如果检查了所有的规则都不匹配，那么该包将被丢弃。这样一个含连接复位（RST）的网络协议包，将被发送到远端的主机。要是该数据包通过防火墙的规则检查被防火墙接收，那么这次连接的信息将被保存在一个连线跟踪信息表中，并表明该包所应有的状态。这个连线跟踪信息表位于内核模式下，其后的网络协议包（也就是在SYN包之后不带SYN标志的数据包）就将和此连线跟踪信息表中的内容进行比较。

根据在信息表中包的信息，来决定该数据包是丢弃还是被接收。利用连线跟踪可以提高防火墙系统的性能，因为每一个数据包首先是与连线跟踪信息表进行比较。只有含有SYN的数据包到来时，它才和规则库进行比较。因为所有的数据包与连线跟踪信息表的比较都是在内核模式下进行的，所以速度很快。

可见连线跟踪的工作原理是，检测每一个有效连接的状态，并根据这些信息决定网络数据包是否能够通过防火墙。该技术在协议栈底层截取数据包，然后分析这些数据包，并且将

当前数据包及其状态信息和前一时刻的数据包及其状态信息进行比较，从而得到该数据包的控制信息，达到保护网络安全的目的。

UDP协议的连线跟踪

UDP协议的连接是无状态的，但仍然可以用类似的方法来维护这些连接。当一个UDP包通过防火墙的规则检查时，这次连接也将被添加到连线跟踪信息表中，并如同TCP一样设置一个连接的超时时间值。在这个时间值内，返回的任何一个源地址、目的地址，以及源端口、目的端口相匹配的数据包都将被允许通过。

包处理过程

数据包进入系统后，在图1中，可以看到有三种可能流程。

• 发往本机上层的包（INPUT链）：NF_IP_PRE_ROUTING→NF_IP_LOCAL_IN。

主机接收到发往本地的IP数据包后，首先经过NF_IP_PRE_ROUTING，系统做出路由决定，决定把该数据包发往本地，数据包经过NF_IP_LOCAL_IN后被应用程序所接收。

• 由本机转发的包（FORWARD链）：NF_IP_PRE_ROUTING->NF_IP_FORWARD→NF_IP_POST_ROUTING→NF_IP_FORWARD→NF_IP_POST_ROUTING。

主机接收到需要它转发的数据包后，首先经过NF_IP_PRE_ROUTING，如果没有任何钩子函数的话，系统将做出路由决定，决定该数据包的走向。然后数据包会经过NF_IP_FORWARD、NF_IP_POST_ROUTING，离开系统并被发送到网络上。

• 从本机发出的包（OUTPUT链）：NF_IP_LOCAL_OUT->NF_IP_POST_ROUTING。

当本地产生数据包时，该数据包首先经过NF_IP_LOCAL_OUT，数据包经过NF_IP_POST_ROUTING后被转发到网络中。

在每一条路径上，连线跟踪、包过滤和地址转换都注册了相应的函数，并创建与之相关的数据结构，完成其功能。每条路径上功能点的顺序如图2所示。



图2 路径功能顺序

DNAT代表目的地址转换，SNAT代表源地址转换，FILTER代表包过滤。可以看到，CONNTRACK在路径上出现了两次，其作用是，第一个点上它创建连线跟踪的结构，这个结构会在后面的地址转换和包过滤中被使用，在第二个点上它将连线跟踪的结构加到系统的连接表中。通过查看ip_conntrack_core.c源代码可以看出fp_conntrack_hash这一双向链表是可供外部访问的，这为后面编写内核代码来读取该表“窥视”表中数据来获取所要的信息提供了可能。

限制P2P连接数内核模块

由于Linux的开放与自由，用户可以通过编写内核模块装载到内核中，利用Netfilter的钩子来达到任何目的。由于P2P连接都是TCP或者UDP连接，因此将利用连线跟踪里面“状态”这一概念来手工进行内核模块的编写，在有数据包进来时先遍历内存中的连线跟踪表，查看里面该IP有多少已建立的连接，若已达到所限制的数目，则把刚进来的数据包丢掉，从而实现连接数的限制。不妨假设一个简单的网络拓扑图如图3所示。



图3 网络拓扑图

该Linux服务器已通过iptables工具配置好了NAT以及一些访问策略等，这里说明一下FORWARD链的配置，该链至少要有如下配置：

```
#iptables -t nat -a POSTROUTING -o eth0 -s 192.168.0.0/16 -j MASQUERADE
#iptables -a FORWARD -i eth0 -m state --state ESTABLISHED RELATED -j ACCEPT
#iptables -a FORWARD -s 192.168.0.0/16 -j ACCEPT
#iptables -a FORWARD -j ACCEPT
```

模块流程图如图4所示。

主要代码如下：

```
static struct nf_hook_ops nfho;
/*实施连接限制的IP地址段，以网络字节顺序表示*/
static unsigned long lim_min_ip=0x0201A8C0;/" 192.168.1.2"
static unsigned long lim_max_ip=0xFEFFA8C0;/" 192.168.255.254"
static int lim_conn=20;
/*每IP限制的连接数为20*/
static int count_conntrack_iterate(const struct ip_conntrack_tuple_hash * hash,const unsigned long our_ip,int *
```

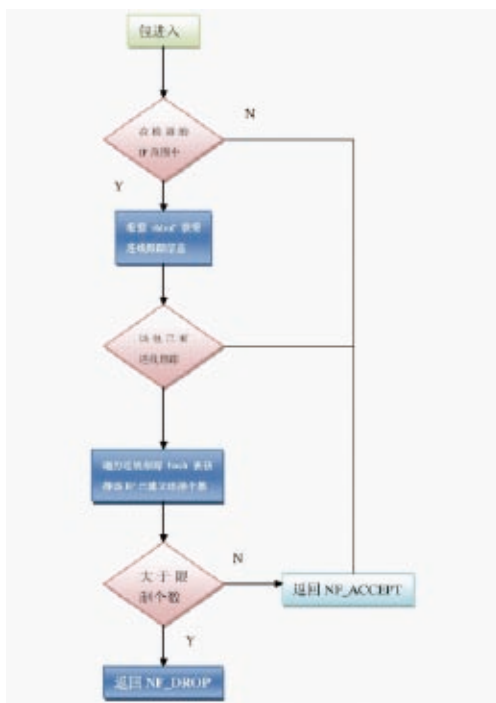



图4 模块流程图

```

pcount) {
    struct ip_conntrack * ct=hash->ctrack;
    int protonum;
    IP_NF_ASSER(hash->ctrack);
    MUST_BE_READ_LOCKED(&ip_conntrack_lock);
    /*只考虑原方向上的连线跟踪*/
    if(DIRECTION(hash))
        return 0;
    protonum=ct->tuphash[IP_CT_DIR_
        ORIGINAL].tuple.dst.protonum;
    if(protonum I=17 && protonum I=6)
        /*如果不是TCP和UDP连接*/
        return 0;
    /*若连线跟踪元组里面的源IP地址为当前数据包的
    IP地址, 则计数加1*/
    if(cur_ip == ct->tuphash [IP_CT_
        DIR_ORIGINAL].tuple.src.ip)
        (*pcount)++;
    return 0;
}

static void count_conntrack(const unsigned
long ip,int* pcount)
{
    unsigned int i;
    READ_LOCK(&ip_conntrack_lock);
    /*遍历整个连线跟踪元组哈希表*/
    for (i=0;i<ip_conntrack_hhtable_size;i++)
    {
        LIST_FIND(&ip_conntrack_hash
        [i],count_conntrack_iterrate,struct ip_
        conntrack_tuple_hash *,ip,pcount);
    }
    READ_UNLOCK
    (&ip_conntrack_lock);
}

/*钩子函数定义*/
unsigned int limitp2p_hook_func(unsigned
int hooknum,struct sk_buff)**skb,const
struct net_device *in,const struct net_

```

```

device *out,int(*okfn)(struct sk_buff *))
{
    struct sk_buff *sb=*skb;
    /*若不在实施连接限制的IP地址段, 交给下一层处理*/
    if(sb->nh.iph->saddr<lim_min_ip||sb-
    >nh.iph->saddr>lim_max_ip)
        return NF_ACCEPT;
    struct ip_conntrack *ct;
    enum ip_conntrack_info ctinfo;
    /*若有连线跟踪说明该连接已建立或者未建立正等
    待对方回应*/
    if(ct)
        return NF_ACCEPT;
    int cur_conn=0;
    /*根据数据包中源IP地址遍历连线跟踪哈希表获得
    已建立连线数*/
    count_conntrack(sb->nh.iph-
    >saddr,&cur_conn);
    if(cur_conn>lim_conn)
        return NF_DROP;
    else
        return NF_ACCEPT;
}

/*模块初始化函数*/
int limitp2p_init_module()
{
    nfho.hook=limitp2p_hook_func;
    /*钩子函数名*/
    nfho.hooknum=NF_IP_PRE_ROUTING;
    /*Netfilter框架在IPv4的检查点*/
    nfho.pf=PF_INET;
    /*采用的协议簇, 这里采用IPv4*/
    nfho.priority=NF_IP_PRI_FIRST;
    /*使编写的函数具有优先调用*/
    /*向Netfilter框架登记钩子函数*/
    nf_register_hook(&nfho);
    return 0;
}

/*模块卸载函数*/
void limitp2p_cleanup_module()
{
    /*向Netfilter框架取消钩子函数登记*/
    nf_unregister_hook(&nfho);
}

module_init(limitp2p_init_module);
module_exit(limitp2p_cleanup_module);

```

本内核模块在Linux 2.6内核下通过编译并能正常工作, Linux提供的基于内核的Netfilter框架, 具有通用性和可扩展性的特点, 使得开发人员可结合自己的需要编写内核模块来对数据包进行处理。文中所述模块只是实现了很简单的功能, 完全可以进行扩充, 例如加上检测的端口等, 或者与iptables相结合, 这样就可以在用户空间通过iptables的参数来配置该模块了。P



李晨光

就职于中国中铁, 资深网络专家, IBM精英讲师, 在IT领域工作十余年, 曾先后从事软件开发、系统及网络管理、信息系统项目集成、网络安全管理等工作。

责任编辑: 董世晓 (dongsx@csdn.net)

【编者按】本文由微软一站式示例代码库团队撰写。微软一站式代码示例库（Microsoft All-In-One Code Framework）是由微软提供的一站式的免费代码示例库。从这里可以免费获得所需的示例代码，解决常见编程问题或提交示例代码需求。微软一站式示例代码库中文官方博客网址为<http://blog.csdn.net/MSCodeSample>。

Azure 和 Bing Maps API 示例经验分享

头疼的Bug，糟糕的代码，崩溃的调试……作为开发人员，遇到上述任何一种情况可能会陷入抓狂。如果能直接获得需要的代码，编程的活儿就会轻松许多。

微软最新推出的一站式示例代码库，让开发人员可以免费获得所需的示例代码或向微软工程师提出示例请求，轻松解决常见的编程问题，大大减轻工作负担。

本文以一个名为AzureBingMaps的示例应用程序为例，分享了一些在开发该示例过程中积累的经验，以期对广大开发人员有所帮助。AzureBingMaps是一个旅游站点管理系统，演示了很多技术，可以认为是一个实际项目。

写这个示例的初衷

在Windows Azure论坛，我们常见到这样的开发人员：他们已经学习了很多开发技术，例如ASP、.NET、Silverlight等，并对这些技术有了较深入的了解。但当他们需要学到的知识应用到实际项目中时，新的问题便产生了。

- 针对特定场景该如何选择平台和技术？
- 不同的技术怎样结合起来使用？
- 如果在使用某项技术的过程中发现了局限性该如何解决？

- 如果必须使用不熟悉的技术该怎么办？

如今的网络技术资源绝大多数都只针对某一种特定的技术，指导人们如何使用一个特定的功能，对那些希望学以致用、开发实际项目的开发人员而言，这远远不够。

鉴于此，我们开始尝试使用微软的各种技

术开发一个相对完整的项目，体会大家可能遇到的问题，从而形成了本文。

选择合适的平台与技术

了解用户需求

在项目开发前，必须了解客户的需求。这项工作的范围很广，但由开发人员负责的部分通常仅限于选择合适的平台与技术。因此作为一个示例，我们省略了与客户访谈以了解需求的过程，直接将非功能性需求定义如下。

- 本系统在旅游旺季需要支持1,000,000个用户同时访问，在非旅游旺季只需要支持1,000个用户同时访问。
- 公司没有自己的数据中心，IT部门最多只能提供3台中档服务器给我们的系统。
- 我们团队对.NET和Visual Studio比较熟悉。
- 本系统对操作系统及网络环境并没有特定的需求。
- 第三方开发人员应该可以针对我们的服务自行开发客户端应用程序。

这些需求也正是我们的客户——Windows Azure论坛上参与讨论的开发人员——常常需要解决的问题。

选择合适的平台

需求明确地指出可伸缩性是必须考虑的因素。为了满足旅游旺季时1,000,000个用户同时访问的需求，我们可能会考虑如下方案。

- 使用一台高性能服务器，然而IT部门明确告诉我们他们只能提供3台中档服务器。
- 采用负载均衡，然而3台中档服务器即使

采用了负载均衡也很难保证满足我们的需求。

- 寻找云计算供应商，将我们的系统部署在外部的数据中心，如果选择的供应商合适，支持多台服务器负载均衡，就能确保满足高并发访问的需求。

于是，我们的需求引领我们考虑选择云计算。然而市场上也有很多云计算供应商，选择哪家最适合呢？这个问题还是要通过需求来解答。

- 在非旅游旺季我们只需要支持1,000个用户同时访问，因此我们选择的供应商必须允许我们随时更改使用计划，例如，旅游旺季租用2,000台服务器，非旅游旺季只租用2台服务器。

- 鉴于项目组对.NET和Visual Studio比较熟悉，我们希望应用现有的知识进行开发，这意味着我们选择的供应商必须支持.NET。

- 既然我们的系统对操作系统和网络环境没有特定的需求，我们就不希望花太多的时间在这些环境配置上。例如，我们不希望手工配置操作系统和安装各种需要的软件；希望即使需要租用2,000台服务器，也可以让项目组致力于应用程序的开发，而不是基础设施的配置。

综上所述，我们发现Windows Azure平台可以满足需求。在Windows Azure平台中，我们可以随时简单通过修改配置文件的方式来选择租用几台服务器，而且理论上可租用的服务器数量确实没有上限。它也完全支持.NET平台，而且操作系统以及常用的软件（例如数据库），也不需要手工配置。

当然，我们承认如上定义正好符合Windows Azure平台的需求，这也是出于我们是针对这个平台撰写示例的考虑。但在实际项目中，大家确实需要考虑上述因素。如果你不需要高度可伸缩性，Windows Azure平台可能就不适合你，毕竟它的价格相对于一般的Web供应商而言是比较高的。如果你对操作系统和网络环境有特定的需求，那么目前Windows Azure平台也不适合你。你应该根据实际需求，寻找合适的平台。

选择合适的技术

在选取技术的过程中，客户需求以及开发团队的经验也是非常重要的。

需求指出第三方开发人员需要针对我们的服务自行开发客户端程序，因此开发服务时我们需要选择一个能让较多客户端平台都接受的技

术，最好是一个国际标准。于是我们决定使用REST。此外，我们的服务需要暴露一些数据给客户端，因此将使用OData。OData是基于REST标准，定义了如何访问数据的一种拓扑，并且被广泛地使用着。我们的开发团队熟悉.NET，于是我们选择在.NET平台上能方便地实现OData的一项技术，也就是WCF Data Services。

在数据存储方面，Windows Azure平台上有两种常见的数据存储服务：Table Storage和SQL Azure。考虑到Table Storage目前还有较多局限性（例如不支持排序），我们决定使用SQL Azure。不过SQL Azure也有自己的局限性，最重要的一点就是目前它不具备Table Storage所提供的自动伸缩功能，也就是说当数据量大的时候，如何确保高效访问数据，是一个问题。不过这个问题也不是特别难以解决，请参考本文“设计可伸缩的数据库”章节寻找解决方案。此外，SQL Azure还支持空间数据（Spatial Data），也就是存放地理信息的数据，我们示例的场景正需要地理信息，所以空间数据也是一个很自然的选择。

Plan My Travel



AzureBingMaps示例代码运行效果截图

至于数据访问，.NET平台提供了Entity Framework，这是一种O/R Mapping的框架，可以让开发人员在不需要考虑如何撰写SQL语句的情况下进行数据访问操作，而将精力专注于面向对象的设计。不过目前Entity Framework对空间数据的支持并不很完美，所以采用它将会给项目带来一定风险。另外一个选择是直接使用SqlConnection以及SqlCommand，但这种方式比较烦琐，而且代码也不易维护。综合考虑，我们决定先做一个简单的原型，尝试将Entity Framework和Spatial Data结合使用，如果在开发该原型的过程中遇上了太多困难，我们将采用SqlConnection的方式。当然最终证明困难并不是很大，于是我们的示例还是采用了Entity Framework。

最后还有客户端，在客户端的技术选择上，我们首先考虑是选择Web还是Desktop。绝大多数情况下，Web应用程序都占据着得天独厚的优势，因为用户不需要安装，甚至不需要下载。当然Web应用程序在用户体验上可能略有不足，不过随着HTML5以及Silverlight的普及，差距也是越来越小了。如今Desktop程序最大的优势在于能够访问更多的系统资源，以及可以更好地支持离线使用。对于我们的场景而言，我们不需要访问特定的系统资源，而且可以暂时不考虑离线访问的状况，所以针对PC类的大型设备我们选择了Web。不过，手机类的设备则是另外一回事。大多数手机浏览器不仅相对而言屏幕较小，而且功能支持也比较少，例如Silverlight一类的插件不受支持，而且也缺乏PC浏览器常见的那种Tabbed View一类的效果。所以如果针对手机设备开发，往往还需要选择该设备直接支持的技术。

至于为何选择AJAX和Silverlight两个PC客户端，以及Windows Phone，就纯粹是出于示例的需要了。还是那句话，如果你的需求不同，你就应该根据需求选择适合于当前项目的技术，而不是生搬硬套。

其他经验分享

设计可伸缩的数据库

Table Storage提供了自动伸缩的功能，是指在同一张表中，数据可能存放在同一台服务器或者不同的服务器上。在Table Storage中，你需要为每条数据指定一个Partition Key。如果两条数据使用相同的Partition Key，那么它们将必然被存放于同一台服务器上。而如果Partition Key不同，那么它们可能会被存放在同一台服务器上，也可能被存放在不同的服务器上，Table Storage将根据数据的使用情况自动决定。

很自然，我们希望当数据需要连续访问时，尽量将它们存储在同一台服务器上，以确保只需要发一个请求就能得到所有数据。但当数据并不需要连续访问时，我们会希望它们尽量存放在不同服务器上，以尽可能地利用负载均衡。所以，Partition Key的选择是非常重要的。

SQL Azure目前并未提供类似的功能，因此我们需要手工来实现。既然SQL Azure和SQL Server一样允许使用复合主键（即多个列共同组成的主键），我们也完全可以采用Partition Key

来决定该数据存放在哪个数据库，而使用Row Key来唯一标志该条记录。

由于SQL Azure将针对每个数据库单独收费，因此还必须考虑不要一次性创建太多的数据库。所以最终我们决定一开始便将所有的数据存放在同一个数据库中，随着数据量的增加，将一部分Partition迁移到其他数据库中。而在数据访问层，我们也提供了一个简单的机制，根据Partition Key来判断该数据应该存放在哪个数据库，以选择合适的连接字符串。你可以参考示例代码的Web Role中的DataService\TravelDataServiceContext.cs中的GetConnectionString方法。当然目前的实现比较简单，因为我们所有的数据都存放在同一个数据库中，然而这种设计对于将来的扩展还是很有帮助的。

排除不确定性：结合Entity Framework使用空间数据

在一开始设计本示例的过程中，我们对于Entity Framework是否能和空间数据（Spatial Data）结合使用抱有质疑。因此我们决定先建一个简单的原型，验证自己的思路是否可行。在大家的实际项目开发过程中，如果遇上类似的不确定性，也可以考虑类似的方案。

在创建原型的过程中，我们参考了大量网络上的技术资源，例如Julie Lerman的博客。最终发现，如果在数据库中对使用空间数据的表创建一个视图，就可以在Entity Framework中查询数据。而要更新数据，则还需要写一些存储过程。不过即使我们直接使用SqlConnection和SqlCommand，不管使用存储过程还是嵌入在程序中，还是需要撰写SQL语句。因此最终我们决定使用Entity Framework，以充分利用O/R Mapping提供的优势。

另一方面，假设事实证明开发这个原型可能要花很长时间很多精力，我们最终会决定直接使用SqlConnection的模式，因为我们知道这种模式是肯定可行的，只是开发起来不太方便。原型帮助我们排除了不确定性。

解决技术的局限：结合WCF Data Services与Entity Framework自定义属性

我们在设计Entity Framework的对象时，考虑到了空间数据很适合于存储地理信息，但对外暴露的接口可能就不太合适了。我们希望我们的服务暴露一个简单易用的Data Contract，因

此在EDM模型中自定义了诸如经度和纬度的属性，而不是直接使用Entity Framework中的空间数据相关属性。

后来我们开始着手开发WCF Data Services。我们原先以为这项工作很简单，因为WCF Data Services内置就支持Entity Framework Provider，只需要简单配置一下就能拥有一个功能强大的服务。然而事实证明我们的想法是错误的：如果你使用Entity Framework Provider，就不能使用自定义属性，只能使用在EDM模型中定义的属性。

这是在项目开发到一半时发现的问题，是一个技术的局限。遇上这种情况，通常应该考虑以下几步。

- 确定是技术的局限，而非知识的不足。
- 若是技术的局限，寻找可行的解决方案。
- 如果找不到可行的方案，评估该功能对于项目是否至关重要。
- 如果不是至关重要，可以暂时先不实现这一功能，将它留到下一个Milestone或版本。
- 如果是至关重要的功能，就必须重新评估最初选择的技术。

作为微软组建运作的团队可以很方便地联系WCF Data Services产品组，并且确认这的确是该技术目前的一个局限性。同时产品组也给我们提供了解决的建议，那就是使用Reflection Provider，而不是Entity Framework Provider。也就是把EDM模型视为一个普通的CLR类型。WCF Data Services支持自定义CLR类型作为数据源，虽然这意味着你需要多写一些代码（尤其是为了支持增删改），但这确实是一个可行的解决方案，而且从架构角度来讲，采用自定义类型也进一步分离了数据访问与服务接口，从而保证了EDM模型更适用于数据访问，服务暴露的数据接口更适合客户端访问。这就是我们最终采用的手段。

如果在开发实际项目的过程中遇到类似的情况，就需要调查你遇上的问题是不是技术的局限性，以及可行的解决方案。如果真的没有任何可行的解决方案，请参考上述建议，评估是将该功能留到下个版本，还是选择另外一项技术。

使用尚不熟悉的技术：HTML客户端的实现

如果说团队只有一名开发人员，不能保证对所有需要使用的技术都很熟悉。在遇上这种状况时，每个人都有自己的解决方案。以下是

我们的经验。

我们的开发人员对于jQuery和CSS不能说是很熟悉，因此在开发HTML客户端时遇上了一些困难。这时首先做的一件事是优先开发Silverlight客户端，毕竟Silverlight是我们所熟悉的。这样做的好处是：虽然技术不同，但是HTML和Silverlight客户端还是有很多共同性的。开发Silverlight客户端有助于进一步认识到我们需要什么样的结果，以及每个功能的大致实现思路。如果大家的需求中没有要求开发另外一个客户端，那么也可以考虑先用自己熟悉的技术做一个类似的原型。

如果你的需求不同，你就应该根据需求选择适合于当前项目的技术，而不是生搬硬套。

由于时间局限，我们无法系统地学习jQuery和CSS，但是既然了解了我们需要哪些功能，就可以针对这些方面进行专门学习。例如知道需要AJAX，于是就调查jQuery中实现AJAX的方法，包括Cross Domain时所需要的JSONP的使用。我们知道不需要太多的jQuery插件，因此可以暂时不学习那部分。此外，由于jQuery Template的前身就是ASP.NET AJAX Template，因此我们也可以利用事先已经拥有的知识，使用jQuery Template方便地展示结构化数据，而不需要写太多的代码来操纵DOM。

在学习了一定基础知识之后，就可以开始使用了。当然这个过程中还是遇上了不少问题，不过我们也有同事对jQuery和CSS很了解，因此在那些同事的帮助下，项目进展也算顺利。

结论

以上就是我们在开发AzureBingMaps示例过程中所遇的问题。如果大家对这个示例感兴趣或者想了解更多的示例代码的获取方式，可以访问微软一站式示例代码库的网址<http://1code.codeplex.com/releases/view/59639#DownloadId=201758>，或者参考介绍该项目实现细节的博客系列文章：<http://blogs.msdn.com/b/windows-azure-support/archive/2010/08/11/bring-the-clouds-together-azure-bing-maps.aspx>。📖

责任编辑：董世晓（dongsx@csdn.net）

在调试器中看Win7打电话回家(下)



主持人：张银奎
《软件调试》一书作者，从事软件开发和研究10余年，对IA-32架构、操作系统内核、虚拟技术，尤其对软件调试有较深入的研究。翻译（合译）作品包括《数据挖掘原理》、《机器学习》、《人工智能：复杂问题求解的结构和策略》、《观止——微软创建NT和未来的夺命狂奔》等。

书接上回，本期我们继续讲Win7如何“打电话回家”。上一期介绍了打电话的44种情况、发起打电话动作的方法，以及开始打电话的过程。我们是以WPD预定的电话任务为例来介绍的。简单来说，任务调度器会在预定条件满足时启动RunDll32.exe，执行portabledeviceapi.dll中的WpdPerformSqmUploadW函数。接下来，WpdPerformSqmUploadW函数会动态加载SqmApi.dll——封装好的提供电话功能的公共模块，然后使用这个模块中的函数通过互联网连接远程的SQM服务器，电话接通后会将要“发回家”的SQM文件放入队列，然后启动一个工作线程来传递数据。那么放入队列的SQM文件都包含哪些内容呢？本期就介绍这个问题，让我们从SQM文件说起。

SQM文件

Win7“打电话”时传递的通常是SQM文件，这是一种二进制格式的文件，只有开始的5个字节是可读的ASCII码（图1），内容为MSQMx，很可能是Microsoft Software/Service Quality Monitoring Extension的缩写。



图1 显示SQM文件

启动一个具有管理员权限的控制台窗口（cmd.exe），在其中执行如下命令：

```
Cd \  
Dir *.sqm /s
```

可以遍历出当前系统中位于系统盘上的所有SQM文件，图2是在我的机器上执行的结果，

共有16个SQM文件，总大小约为128KB。总体来说，SQM文件通常不是很大，图2中显示的几个都是一两百个字节或者1K多字节，这与使用二进制文件格式有关，可能是设计时就考虑到了文件要小，“长话短说”，传输起来方便快捷，不要啰哩啰唆。

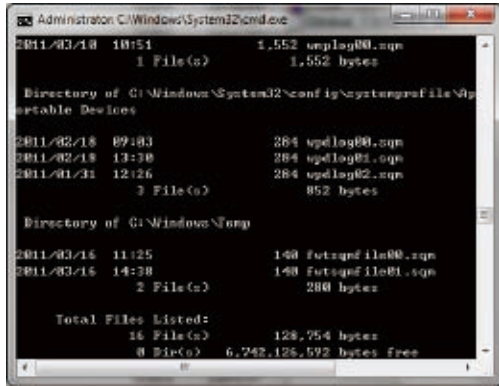


图2 列出系统中的SQM文件

值得说明的是，每次打电话成功后，Win7都会把已经使用的SQM文件删除，因此上面看到的SQM文件都是还没有使用过的，或者说以后打电话时要传递的。图3中的栈回溯显示了上传SQM文件的工作线程在上传结束后调用DeleteFile API来删除文件的情景。



图3 上传后删除SQM文件

SQM集装箱

SQM文件是如何产生的呢？简单来说是由一个名为Windows SQM集装箱（Windows SQM

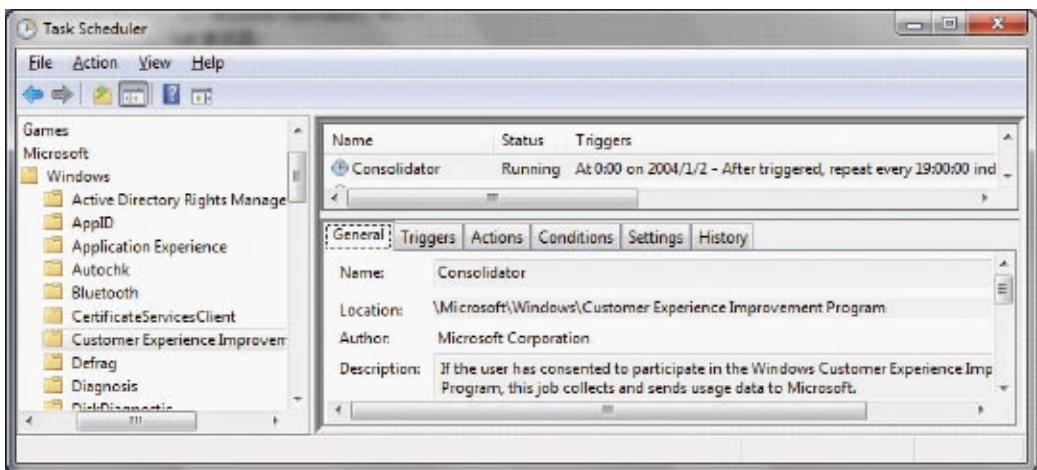


图4 自动启动SQM集合器的预定任务

Consolidator) 的小程序生成的。这个程序的执行文件是wsqmcons.exe，位于系统目录 (C:\Windows\System32) 中。

在任务调度器的\Microsoft\Windows\Customer Experience Improvement Program项目中，我们可以看到一个名为Consolidator的任务 (图4)，它就是用来定期启动SQM集合器的。

因为Consolidator任务被定义成以系统用户 (SYSTEM) 运行，所以调度器会在会话0中启动SQM集合器。因为会话0是不可见的，所以如果使用上期提到的自动附加调试器的方法，那么调试器也会出现在会话0中，尽管启动了，但是我们却看不到，无法操作。如果发现wsqmcons进程起来后，再在会话1中附加调试器，因为这个进程启动后很快退出，所以也很难成功。解决这个问题的方法是使用“远程”调试。具体来说，先在注册表的HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options表键下建立一个名为wsqmcons.exe的子键，然后增加一个名为Debugger的字符串类型的键值，并设置为：

```
c:\wdbg64\windbg.exe -server npipe:pipe=sqm
```

其中的WinDBG文件路径要修改成实际的路径。后面的命令行参数是告诉WinDBG启动后立即启动远程调试服务，连接这个服务的方式是名为sqm的命名管道。

做好以上设置后，手动运行Consolidator任务 (点击右键弹出快捷菜单选择Run)，然后Ctrl + Esc唤出任务管理器，显示所有线程，确认wsqmcons.exe进程已经启动，而且有一

个WinDBG进程运行在会话0中。而后再启动一个WinDBG，选择File→Connect to Remote Session，输入如下连接信息：

```
npipe:Pipe=sqm,Server=127.0.0.1
```

其中的127.0.0.1也可以换成本机的机器名。点击确认后，当前的WinDBG会连接到会话0中的WinDBG，显示出被调试进程的信息，此后便可以输入命令调试了。

可以执行bp wsqmcons!WinMain命令在wsqmcons的入口设置一个断点，然后慢慢跟踪。如果觉得太慢，也可以在感兴趣的地点设置断点进行“中途拦截”。考虑到wsqmcons会创建文件，所以执行bp kernel32!CreateFileW埋下断点，然后执行g命令恢复进程执行。断点命中后，执行k命令观察栈回溯，如果对当时执行的操作不感兴趣，那么便执行g命令放过，直到看到类似图5所示的栈回溯。

```
Call Site
kernel32!CreateFileW
kernel32!CreateFileWImplementation+0x7d
wsqmcons!CSqmFileWriter::WriteBinaryFile+0x592
wsqmcons!CSqmSessionManager::SaveToFile+0x4f
wsqmcons!CSqmSessionManager::EndSqmSession+0x2e2
wsqmcons!CSqmConsolidator::EndSqmSession+0xe3
wsqmcons!CSqmConsolidator::SqmEventEndSession+0x162
wsqmcons!CSqmConsolidator::SqmEventCallbackProc+0x1dd
ADVAPI32!EtwpDoEventCallbacks+0x3e
ADVAPI32!EtwpProcessTraceLogEx+0x9a
ADVAPI32!EtwpProcessTraceLog+0xf6
ADVAPI32!ProcessTrace+0x204
wsqmcons!CSqmConsolidator::ProcessSQMEvents+0xf4
wsqmcons!CSqmConsolidator::Process+0x9a3
wsqmcons!WinMain+0x3c7
```

图5 创建SQM文件

我们知道CreateFile API的第一个参数就是文件名，因此观察第一个参数便可以知道创建的是何文件。对于x86的系统可以使用du @ebp+8这样的命令。对于x64系统，可以执行du @ECX命令，即：


```
0:000> du @ECX
00000000'00299970 "C:\ProgramData\Microsoft\Windows"
00000000'002999b0 "\Sqm\Upload\Private_401412_00.sq"
00000000'002999f0 "x"
```

浏览到这个文件目录，还没有这个文件，看来SQM集装器正在创建一个新的SQM文件。从图5中的函数名我们也可以看出，名为CSessionBlock::SaveToFile的方法在调用CSqmFileWriter::WriteBinaryFile方法。执行gu命令，再看一下这个文件目录，可以发现一个0字节长的文件。对kernel32!WriteFile设置一个断点，恢复执行，即便会命中，观察第二个参数所代表的指针，可以看到正在写SQM文件的文件签名：

```
0:000> db @edx
00000000'002c1110 4d 53 51 4d 78
00 00 00-00 00 00 00 f9 8f 8e 81
MSQMX.....
...
```

如果不想花更多时间逐一跟踪，那么可以执行如下命令，设置一个监视点：

```
bp kernel32!CreateFileW "dU @ECX;kn;gc"
```

于是可以看到SQM集装器可能产生很多个SQM文件，图6是在我的机器上执行这个进程后，观察Upload目录时的情景，共有132个SQM文件，准备就绪，“整装待发”。这些文件大小不等，大的有185KB，小的不到1KB。

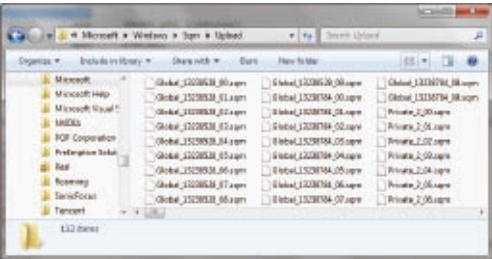


图6 “整装待发”的SQM文件

一下子见到这么多SQM文件的机会并不是很多，正常情况下产生几个，发出去便删掉，不会有这么多。就本例来说，是因为我在写上期文章时做了上面提到的自动启动调试器设置，每次wsqmcons进程起来后，都会中断到调试器中，停止不动。图6中的这些文件是积攒了差不多一个月的内容。

SQM聊天室

看了上面的介绍后，大家可能会问，SQM集装器是从哪里拿到信息写到SQM中去的呢？说来话长，但只能长话短说了。执行perfmon打开性能监视器，浏览到图7所示的Event Trace Sessions分支，会看到右侧有一个名为

SQMLogger的会话正在运行。

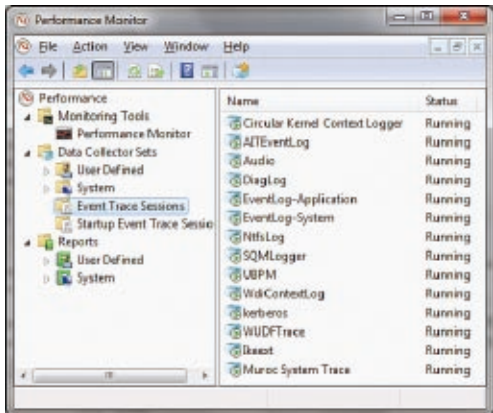


图7 观察SQM日志会话

熟悉Windows的朋友知道从Windows 2000开始Windows便建立了一种名为ETW（Event Trace for Windows）的事件追踪机制。这种机制是基于“提供者—消耗者（Provider/Consumer）”设计模式，具有非常好的灵活性，使用二进制方式记录信息，信息先写在专门的内存缓冲区中，相当高效，核心实现来自Windows内核，有API供用户态和内核态的其他模块使用。有《软件调试》的朋友可以阅读第16章有更详细的介绍。每个ETW会话好像一个聊天室，提供者可以向里面发言，消耗者可以从中读取信息。而SQMLogger会话就是专门供“大家”交流SQM事件使用的。它是以自动记录器的方式登记在注册表中的（图8），保证每次系统启动时，内核初始化时就会创建这个“聊天室”，即使启动过程中的事件也可以记录下来。

图8中SQMLogger表键的一系列子键代表的是这个会话的提供者，也就是向这个聊天室里发信息的“人们”。在图7中双击右侧列表中的SQMLogger条目，弹出图9所示的对话框，可以看到更友好的提供者列表。

观察对话框的Trace Session页，Stream mode为文件（File），意思是这个会话的内容会定期冲转到文件中，选择Directory页，可以看到用于存放文件的目录是：C:\(Windows\system32\LogFiles\SQM。

观察文件页，可以看到当前使用的文件名：SQMLogger_2011-3-16-11-57-52_0。

有了这个基础后，再观察图5中的栈回溯，其中的ProcessTrace就是利用ETW API让系统逐

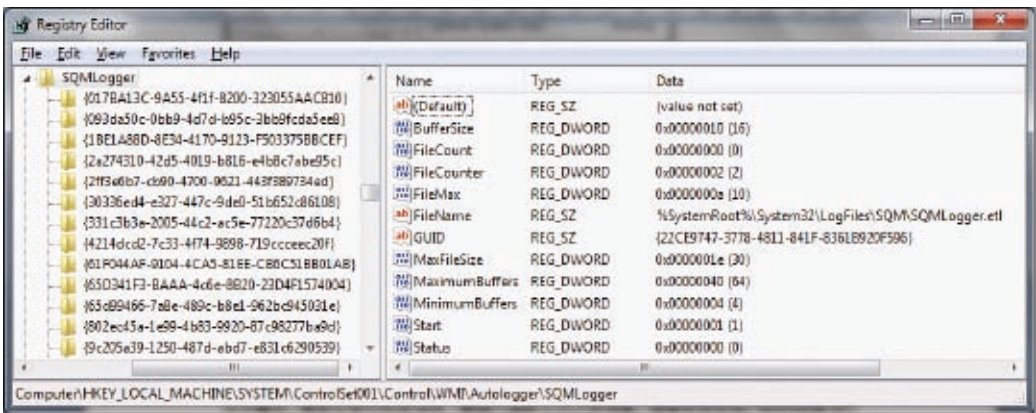


图8 SQMLogger在注册表中的参数

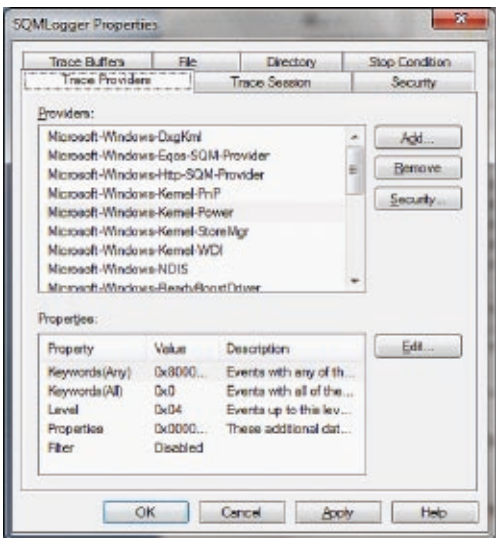


图9 观察SQM日志会话

条回放记录在SQMLogger会话中的事件，然后把属于同一个SQM会话的内容写到一个SQM文件中。可以把这里的一个SQM会话理解成一篇日记，它可能包含多个事件，SQM集装器会把属于一篇日记的事件收集在一起，整理后写到一个SQM文件中，这样就把本来记录在一起的很多类事件分离成了专一的SQM文件。

时常写日记

最后再来看一下SQM事件是如何写到SQM聊天室（SQMLogger会话）的。简单来说，有多种方法，无论是内核模块或者用户态模块都可以直接调用ETW函数，然后写事件到SQMLogger会话。对于用户态模块，更简便的方法是调用NTDLL提供的SQM函数，比如ntdll!WinSqmStartSession、ntdll!WinSqmSetDWORD、ntdll!WinSqmIncre-

mentDWORD、ntdll!WinSqmEndSession等。图10所示的栈回溯便是资源管理器进程中调用SQM API“写日记”的过程。

触发图10中动作的原因是我在任务栏上点击了程序图标以启动新的进程。在栈回溯中可以看到CreateProcess（创建进程）的字样，其中的PcaCheckEnablePca中的Pca是Program Compatibility Assistant的缩写，意思是程序兼容助手。PCA是系统中支持应用程序兼容的一系列设施的总称，因此可以推测这个日记是在记录笔者系统中PCA的启用情况。事实上，当我们调试跟踪Win7的系统模块时，经常会遇到这样的写日记代码。

```
Call Site
ntdll!WinSqmIncrementDWORD
SHELL32!PcaCheckEnablePca+0x42
SHELL32!_SHCCreateProcess+0x50c
SHELL32!CExecuteApplication::_CreateProcess+0x15f
SHELL32!CExecuteApplication::_TryCreateProcess+0x126
SHELL32!CExecuteApplication::_DoApplication+0x198
SHELL32!CExecuteApplication::_Execute+0x3e
SHELL32!CExecuteAssociation::_DoCommand+0xb0
SHELL32!CExecuteAssociation::_Execute+0xbf
SHELL32!CRegDataDrivenCommand::_Invoke+0x10d
SHELL32!CRegistryVerbsContextMenu::_Execute+0x77
SHELL32!CRegistryVerbsContextMenu::_InvokeCommand+0x102
SHELL32!HDXA_LetHandlerProcessCommandEx+0x144
SHELL32!CDefFolderMenu::_InvokeCommand+0x254
SHELL32!CShellLink::_InvokeDirect+0x204
SHELL32!CShellLink::_ResolveAndInvoke+0x1de
SHELL32!CShellLink::_InvokeCommand+0xe3
SHELL32!HDXA_LetHandlerProcessCommandEx+0x144
SHELL32!CDefFolderMenu::_InvokeCommand+0x254
SHLWAPI!SHInvokeCommandOnContextMenu+0x1c8
SHLWAPI!SHInvokeCommandWithFlagsAndSite+0xa0
Explorer!CTaskBand::CLauncherTask::_Launch+0x19a
Explorer!CTaskBand::CLauncherTask::_s_ThreadProc+0x12
```

图10 调用WinSqm函数写日记

时间不早了，已经夜里11点，快到Win7打电话的时候了，今晚的电话也许比较长，因为有很多SQM文件要发，或许还会增加一条内容——“有人偷听电话了。”📞

本期问题：

上期答案是：MSQM，只要按本期本章中的介绍搜索到sqm文件，然后用编辑器或者type命令观察即可。

本期问题是：请大家使用这两期文章中提到过的程序工具观察Win7“打电话”的记录，看其都在哪些时候执行了“打电话”有关的动作，结果如何？

编者说明：

- 投稿信箱：
contest@csdn.net
- 联系方式写在一个单独的TXT文件里，包括以下几项：
 - 1) 姓名
 - 2) 工作单位或学校
 - 3) 电话联系方式
 - 4) 邮寄地址
 - 5) E-mail地址
- 解答提交时间，最好早于当月15日。

Android自动化测试之道

文 / 杨丰盛

随着越来越多的手机厂商、运营商、开发商加入Android阵营，带来了各种美化和定制的Android系统，同时Android Market的应用数量增长迅速。这都离不开测试，Android自动化测试的需求愈加迫切。

测试主要分为白盒测试和黑盒测试两大类。Android虽然在测试方面还不够完善，但也提供了各种途径来完成相应的测试模块。白盒测试通常需要配合软件的源码来进行，测试人员需根据产品的功能和性能等需求编写测试用例，并根据测试用例逐一进行测试。这里我们主要探讨在没有源码的情况下如何进行黑盒测试。既然没有源码，那么我们就需要使用Android所提供的工具包来获取应用程序上的各种UI元素，并向其发送各种操作事件。此外，我们还可以借助各种开源项目，比如ASE等。下面分别介绍几种目前最可行的自动化测试解决方案。

Android兼容性测试

Android的各个版本之间都有不同程度的差异，因此Google专门针对各个版本（Android 2.1以上）提供了兼容性测试工具CTS（Compatibility Test Suite），其中共有两万多个测试用例，包括功能测试和性能测试等。CTS最初只对OHA联盟开放，后来便和Android开源项目一起发布了，并且所有手持设备都必须通过Android兼容性测试，才能确保在Android上开发的应用程序能够在Android设备上运行，才能使用Android Market。

要使用CTS进行兼容性测试，就必须先下载和编译CTS，下载方式可以选择下载完整的Android源码（位于\$ANDROID/cts目录中）和只下载CTS源码（注：CTS源码下载地址为git://android.git.kernel.org/platform/cts.git）；然后可以使用如下命令来编译CTS：

```
//配置选项
$ build/envsetup.sh
//编译cts
$ make cts
```

编译完成之后会将结果输出到“\$ANDROID/out/host/linux-x86/”目录中，然后可以启动模拟器或手持设备（需使用root权限来操作），并进入其bin目录。输入“./cts”命令来启动CTS测试，显示CTS测试版本和设备连接状态，这时可以输入“help”命令查看CTS的操作帮助信息。

如果输入“ls --plan”命令可以查看所有的plan（注：plan文件夹为cts/android-cts/repository/plans），在其中可以看到8个测试基类，比如Android.xml、AppSecurity.xml、CTS.xml等。其中所有基类里包含的package都给出了相应的uri，cts将根据这些uri去测试每个基类里的package，可以输入如下一些命令来测试某些package：

```
//测试所有cts测试用例
cts_host > start --plan CTS
//测试Android API
cts_host > start --plan Android
//测试Java核心库
cts_host > start --plan Java
```

通常在测试中很可能会出现错误，这时若需要知道是在哪一个plan处出错以及出错的原因，可以使用“-plan”、“-p”、“-t”、“-e”、“-w”等参数来指定测试某个用例，可以节约调试时间，如下命令可以指定测试AlertDialog：

```
cts_host > start --plan Android -p
android.app -t android.app.cts.
AlertDialogTest#testAlertDialog
```

当测试结束后，可以通过“ls -r”命令来查看测试的结果（注：CTS测试结果路径cts/android-cts/repository/results），如果觉得现有的测试用例不够完整，也可以根据现有的测试用例来添加自己编写测试用例，但需要注意每次添加或修改CTS任何模块都需要重新编译才能进行测试。

CTS主程序“bin/cts”是一个用sdb来测试的脚本程序，所有的测试用例都使用Java语言编写，其主程序为TestHost.java（注：CTS主函数为cts/

tools/host/src/com/android/cts/TestHost.java)。CTS是基于Android Instrumentation的测试，因此又是基于JUnit的测试，实际上也就是一套单元测试用例。

Android Instrumentation测试

Android整合了JUnit测试框架和Instrumentation机制。JUnit是广泛应用于Java程序开发中的单元测试框架，定义了特定格式的方法作为测试用例，提供TestSuite测试套件和TestRunner测试执行器分别用于组织测试用例和运行测试用例，并反馈测试结果。Instrumentation则是一种操作系统和上层应用程序之间的监控机制。当应用程序运行时，若Instrumentation设置为开启，它将在应用程序运行前初始化，与应用程序运行在同一个进程中，监测应用程序与系统之间的交互，并对应用程序进行一定的控制，例如可以监控应用中各Activity的调用情况、启动和结束Activity、对Activity生命周期中的各个状态进行切换，甚至可以向Activity发送按键事件和字符，图1展示了Android中Instrumentation测试的构架。



图1 Android Instrumentation构架图

Instrumentation和Activity有点类似，不同的是Activity需要一个界面，而Instrumentation并不需要，可以将它理解为一种没有图形界面的、具有启动能力的、用于监控其他类（用Target Package声明）的工具类。

在Android中，有关测试的类都继承自TestCase抽象类，它实际上就是对JUnit的Instrumentation扩展，其中又包括了InstrumentationTestCase和AndroidTestCase，图2展示了详细的Android Instrumentation测试框架。

当需要测试某个应用程序时，可以直接从上面的类中派生一个子类即可，例如，要测试一个HelloWorld程序，其Activity名称为HelloWorld，就可以创建一个继承自ActivityTestCase的子类HelloWorldTest来对其Activity进行测试；其超类

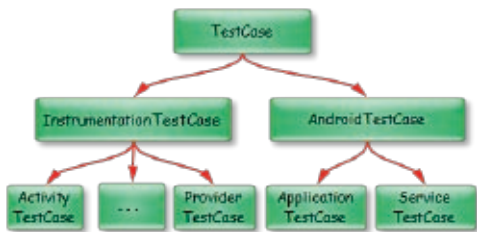


图2 Android Instrumentation测试框架

中提供了一些受保护的方法，例如setUp()可以用来初始设置，如启动一个Activity、初始化资源等；tearDown()可以用来垃圾清理与资源回收。

编写完测试用例还需要修改AndroidManifest.xml文件，首先需要引入测试库，并指明需要被测试的目标package，代码如下：

```
<uses-library android:name="android.test.runner"/>
<instrumentation android:targetPackage="com.yarin.HelloWorld"
    android:name="android.test.InstrumentationTestRunner"/>
```

如果使用Eclipse作为开发环境，便可以将Android JUnit Test工具和Eclipse进行继承，并在LogCat工具中打印日志信息。

Android Instrumentation基本上可以测试大部分Android API，但也有部分API不能被获取，如Toas等；当遇到一些复杂的交互性测试比较难以实现，这时可以考虑模拟键盘、鼠标事件以及使用图像识别技术来完成自动化测试。

Android在操作系统层提供将键盘事件转化为应用程序动作的能力，一个事件的成功触响应需要如下几个步骤：首先，Window Manager接受来自Linux键盘驱动的按键事件；其次，Window Manager把scancode映射成keycode；再次，Window Manager把scancode和keycode都发给应用程序；最后，应用程序对按键事件进行响应。

因此可以通过“adb shell input/sendevent”命令来向设备发送键盘和触摸操作事件。以下命令所发生的事件为在坐标（00000045，00000032）处点击屏幕，并且保持按下状态：

```
adb shell sendevent /dev/input/event0:
0003 0000 00000045
adb shell sendevent /dev/input/event0:
0003 0001 00000032
adb shell sendevent /dev/input/event0:
0001 014a 00000001
```

要实现自动化测试，就需要把上述操作都通过程序来自动完成，这时可以使用Android工具包中的hierarchyviewer工具（hierarchyviewer.jar）或者其他脚本程序。同时Instrumentation中也提

供了以send开头的函数接口来实现模拟键盘鼠标，如下代码所示：

```
//发送指定KeyCode的按键
sendCharacterSync(int keyCode)
//发送指定KeyCode的按键
sendKeyDownSync(int key)
//模拟Touch事件
sendPointerSync(MotionEvent event)
//发送字符串
sendStringSync(String text)
```

如果使用图像识别技术进行检测，可以使用如下两种方案进行检测：

- 利用系统viewserver中提供的接口获得当前界面上控件的属性信息，通过对比测试前后属性信息的变化进行结果校验；
- 利用ddmlib包中提供的接口自动截取测试前后的屏幕截图，通过对比两幅截图相同位置的图像信息是否存在差异来进行结果校验。

无论是使用事件触发还是图像识别技术进行检测，都还需要完成大量的测试用例编写，以便能更好、更高效地完成自动化测试。

Android 压力测试

压力测试是任何一个应用程序和操作系统都必须完成的测试，Android开发套件中专门为开发者提供了一个用来做压力测试的工具Monkey。它可以在模拟器和设备上运行，其原理是通过Monkey产生随机脉冲，并随机产生大量事件对软件的负载能力进行测试。

Monkey是一个命令行工具，使用方法非常简单。它提供四个基本的配置，分别是：基本配置选项、运行约束选项（如设置Monkey只对单个包进行测试）、事件类型和频率、调试选项。

这里主要对其调试选项进行分析，当Monkey运行时，它会生成事件，并把它们发给系统。同时，还会对测试中的系统进行监测，对下列三种情况进行特殊处理。

- 如果限定了Monkey运行在一个或几个特定的包上，那么它会监测试图转到其他包的操作，并对其进行阻止。
- 如果应用程序崩溃或接收到任何失控异常，Monkey将停止并报错。
- 如果应用程序产生了应用程序不响应的错误，Monkey将会停止并报错。

Monkey工具作为Google官方提供的压力测试工具，其效率非常高，无论是应用程序还是系统都可以使用它来测试其负载能力。

ASE测试

ASE（Android Scripting Environment）是Google针对Android发布的一个小工具。它将脚本语言带入Android，允许用户编辑和执行脚本，直接在Android设备上运行交互式解释器。脚本能大幅度简化任务界面，目前支持Python、Perl、JRuby、Lua、BeanShell、JavaScript、Tcl、shell，以后将会增加更多的语言的支持。因此，就可以方便快捷地调用Android系统中大部分的API，根据官方发布的消息，它可以通过简便的接口快速实现以下功能：Handle Intents、启动Activities、发送通话请求、发送文字信息、扫描条形码、获取地理信息和返回设备传感器状态信息、使用语音输出文字信息（TTS）。

熟悉上述任何一种脚本语言的测试人员都可以选择使用ASE开源项目来进行测试，操作过程如下：下载ASE（注：下载地址为<http://code.google.com/p/android-scripting/>）、安装ASE、安装解释器（打开ASE应用，选择自己熟悉的脚本解释器进行在线安装）、编写测试用例（编写脚本程序来操作Android API及其应用程序）。

如果在测试过程中，ASE提供的API接口不能满足其需求时，可以对ASE项目进行扩展；因为ASE项目的可扩展性非常强，和Android Instrumentation一样，只需要从AndroidFacade和AndroidProxy中进行派生即可。

总结

本文介绍了Android上常用的几种自动化测试解决方案，对各方案的原理和使用方法进行了介绍，基本可以完成Android各功能模块的测试。但真正进行自动化测试时，还需要对各种方案进行整合与完善，以发挥出更高的效率。如果上面所介绍的不能满足测试需求，可以考虑使用其他测试方案以及测试管理工具，如Robotium、AutoIT、MantisBT等。



杨丰盛

佳玩无线联合创始人，精通Java、C/C++等语言及J2ME、Android、iPhone等平台，曾领导和参与《三国群英传》、《大航海传奇》等游戏的开发，著有《Android应用开发揭秘》。

责任编辑：董世晓（dongsx@csdn.net）

volatile与多线程的那些事儿

文 / 陈冠诚

在进行多线程编程时，我们需要慎重使用volatile关键字，本文分析了C/C++和Java多线程程序中使用volatile关键字的一些例子，帮助大家更好地理解和使用volatile。

随着多核的日益普及，越来越多的程序将通过多线程并行化的方式来提升性能。然而，编写正确的多线程程序一直是一件非常困难的事情，volatile关键字的使用就是其中一个典型的例子。由于历史原因，volatile在C/C++和Java/.NET中产生了两种完全不同的用法，从而导致了许许多多的误解。本文的主要目的是分析volatile关键字在多线程程序中的一些常见错误以及正确用法，以帮助大家更好地利用volatile的特性。

C/C++中的volatile一般不能用于多线程同步

在C/C++中，如果把一个变量声明为volatile，就相当于告诉编译器这个变量是“易变的”，它随时可能在别处（例如其他硬件等）被修改，所以编译器不能对它做任何优化：即每次读写该变量时都必须对其内存地址直接进行操作，并且所有对该变量的操作都必须严格按照程序中规定的顺序执行^[1]。举例来说，编译器常做的一种性能优化就是把需要频繁读取的变量缓存到寄存器中，以提升访问速度。但如果该变量的值随时可能在片外被改变的话（例如另一个设备可能更改它的值），那么就有可能出现被缓存的值并不是该变量的最新值的情况，从而出现运行错误。在这种情况下就需要用volatile关键字来修饰这个变量，以确保编译器不会对该变量的读写操作进行任何缓存优化。另一个例子就是内存映射I/O（MMIO）操作。以如下代码为例：

```
int *p = get_io_address();
int a, b;
a = *p;
b = *p;
```

p是一个指向硬件I/O端口的指针，该端口的

值在每进行一次读操作后都会变化。这个程序连续对该端口进行两次读取操作以将两个不同的值分别赋给a和b。如果不把a和b声明为volatile的话，编译器可能会“自作聪明”地认为两次从p读取的值都是一样的，从而把“b = *p”优化成“b = a”，最终导致程序出错。

虽然C/C++中的volatile关键字对这种“易变”的读写操作能起到一定的保护作用，但它却并不适用于多线程程序中共享变量的同步操作。究其根源，就在于C/C++标准中并未给volatile赋予原子性和顺序性的语义。

原子性

下面举例说明原子性。i++这个看似原子的语句其实由三个操作组成：将该值从其内存地址读取到寄存器中，对寄存器中的值进行加1操作，最后再将新值写回内存中。正因为i++并不是原子的，所以如果有两个线程同时进行i++操作的话仍会产生数据竞跑（Data Race），从而导致i的最终值不等于2。在这种情况下，C/C++中的volatile关键字根本无法对该操作的原子性提供任何保障。

```
volatile int i = 0;
// 线程 1
i++;
// 线程 2
i++;
```

顺序性

不幸的是，现有C/C++标准中的volatile关键字对共享变量操作的顺序性也未提供任何保障。以本文中的dekker算法为例：当两个线程分别执行dekker1和dekker2函数时，该程序通过对flag1/2和turn的读写来实现两个线程对临界区中共享变量gCounter的互斥访问。这个算法的关键

就在于对flag1/2和turn的读操作是在其写操作之后进行的，因此它能保证dekker1和dekker2中对gCounter的操作是互斥的，相当于把gCounter++放到一个临界区中去了。dekker算法如下所示。

```
volatile int flag1 = 0;
volatile int flag2 = 0;
volatile int turn = 1;
volatile int gCounter = 0;
void dekker1() {
    flag1 = 1;
    turn = 2;
    while((flag2 == 1) && (turn == 2)) {}
    // 进入临界区
    gCounter++;
    flag1 = 0; // 离开临界区
}
void dekker2() {
    flag2 = 1;
    turn = 1;
    while((flag1 == 1) && (turn == 1)) {}
    // 进入临界区
    gCounter++;
    flag2 = 0; // 离开临界区
}
```

尽管volatile规定编译器不能对同一个变量的所有操作进行乱序优化，但它却并不能阻止编译器对不同volatile变量间的操作进行乱序优化。例如，编译器可能把dekker1中flag2的读操作提到flag1和turn的写操作之前，从而导致对临界区的互斥访问失效，最终gCounter++操作就会出现数据竞跑现象。事实上，即使编译器没有对这个程序做任何优化，volatile关键字也不能阻止多核CPU对该程序的乱序优化。以常见的x86硬件来说，它可以对不同变量x, y的store x → load y进行乱序优化，把load y操作提到store x操作之前（即先执行load y再执行store x）。这样的话，dekker1中flag2的读操作还是有可能被提到flag1和turn的写操作之前，最终导致错误的计算结果。

那为什么编译器和多核CPU会对多线程程序做这样的乱序优化呢？因为从单线程的视角来看，flag1、flag2和turn的读写操作之间是没有任何依赖关系的，所以编译器/CPU当然可以对它们进行乱序优化以隐藏一部分的内存访问延迟，从而更好地利用CPU里的流水线。换句话说，这样的优化虽从单线程的角度来讲没有错，但却违反了设计这个多线程算法时所期望的多线程语义。要解决这个问题，我们需要自己添加内存栅栏（Memory Barrier）以显式保证顺序性，或者干脆别去实现这样的算法，转而使用类似pthread_mutex_lock这样的加锁操作来实现互斥访问。

综上所述，由于现有的C/C++标准中并没有

对volatile添加原子性和顺序性的语义，所以绝大部分C/C++程序中使用volatile来进行多线程同步的用法都是错误的。其实，我们之所以想用volatile变量进行同步，无非是因为锁、条件变量等方式的开销太大，所以想有一种轻量的、高性能的同步机制。这样的需求就引出了Java、.NET中的volatile关键字，即volatile的另一种用途。

Java/.NET中的volatile可以完成部分多线程同步任务

Java中的volatile关键字

从Java 1.5开始，volatile变量被赋予了顺序性的语义。具体说来有以下两点。

- volatile变量的读操作具有acquire语义，即程序中处于该volatile变量读操作之后的所有其他变量（包括volatile与非volatile变量）的操作都必须在该读操作完成之后执行。
- volatile变量的写操作具有release语义，即程序中处于该volatile变量写操作之后的所有其他变量（包括volatile与非volatile变量）的操作都必须在该写操作之前执行完毕。

在给volatile添加顺序性语义之后，编译器和CPU的乱序优化就受到了一定的限制（注：JVM的乱序优化也受到了同样的限制）。图1中的绿色箭头和红色箭头分别代表了可以进行以及禁止进行乱序优化的方向。举例来说，编译器和CPU可以把位于volatile读操作之前的代码移到volatile读操作之后执行，但是反方向的乱序优化却是被禁止的。

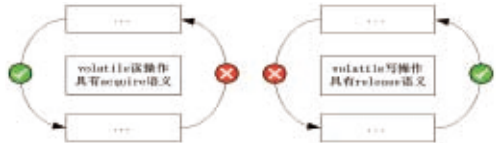


图1 volatile的顺序性语义

在被赋予顺序性语义之后我们就可以在Java中使用volatile关键字来完成一些多线程间的同步操作了。一个典型的用法就是用volatile变量来做多线程间的状态标记。例如在下面这个程序中，两个线程分别执行thread1和thread2，它们通过sleep这个volatile变量来进行同步。

```
volatile int sleep = 1;
int gData = 0;
void thread1() {
    while (sleep) {}
    //读操作一定会在sleep的读操作完成之后才被执行
    System.out.println("gData = " + gData);
}
void thread2() {
```

```

    gData++; //该操作一定会在sleep的写操作之前
              被执行完毕
    sleep = 0;
}

```

例子中的while循环体是对sleep变量的读操作，具有acquire语义，保证了printf中对gData的访问必须在该读操作完成之后才能执行；同理，例子中的sleep=0是对sleep变量的写操作，保证了gData++必须在进行该写操作之前已经执行完毕。所以thread1中打印gData的值一定是在thread2已经执行完gData++之后才会发生，即gData最后的输出值一定为1。

这里有两点值得注意。第一个需要注意的地方是volatile在Java多线程中的适用场景是十分有限的。确切地说，只有对该变量的写操作不依赖于其当前值，而且该变量没有包含在具有其他变量的不变式中时，volatile才能起到正确的同步作用^[2]。如果不能确定使用volatile时多线程程序的正确性，建议大家还是使用Java中的atomic类型或者synchronized关键字来进行多线程间的同步^[3]。另一个需要注意的地方是Java中的volatile关键字并不能保证原子性。因此对于一个volatile变量i来说，i++这样的操作仍然不是原子的。这时我们可以使用类似AtomicInteger中的incrementAndGet()的方法来完成自增操作。

.NET中的volatile关键字

虽然C/C++标准中的volatile关键字没有对顺序性做任何保证，但微软从Visual Studio 2005开始就给volatile关键字添加了与Java一样的顺序性语义^[4]，即：对volatile变量的读操作具有acquire语义，对volatile变量的写操作具有release语义（这个语义属于微软自己添加的扩展，并不在C/C++的标准中，所以仅限于Windows平台）。这使得volatile在.NET中也可以用来完成一些诸如状态标记等多线程同步的任务。同样的，.NET中的volatile关键字也有与Java里类似的局限性，所以在使用时需要格外小心。

C++1x中的volatile也没有顺序性语义

按照时间顺序，C/C++中的volatile关键字是最早诞生的，然后Java给volatile关键字添加了顺序性的新语义，此后微软也给.NET中的volatile添加了顺序性语义。但在即将到来的C++1x新标准中，标准委员会却决定不再给C++中的volatile添加顺序性语义。这主要是因为给volatile添加顺序

性语义会对遗留代码添加额外的性能开销（因为它阻止了编译器/CPU的乱序优化），而且会造成很多不可移植的代码需要被重写，实现代价高昂。因此，C++1x中将转而提供另一种既有原子性又有顺序性语义的atomic类型，我们可以通过它来完成多线程间的同步任务，实现大部分的高性能多线程算法（例如Lock-Free算法）。

总结

总的来说，volatile关键字主要有两种用途：一种是C/C++中用来处理“异常”内存行为（此用途只保证不让编译器对该变量的操作做任何优化，但是对编译器和CPU对不同变量操作间的乱序优化没有任何约束力），另一种是在Java/.NET（包括Visual Studio添加的扩展）中用来实现（功能有限的）多线程间的同步任务。

使用volatile来进行多线程同步是困难的。在使用时，大家一定要仔细弄清楚多线程程序在多核上到底是不是按自己所想的方式被执行，编译器、CPU、JVM等带来的乱序优化到底会不会对程序行为产生影响，特别是对暂时还没有引入多线程语义的C/C++程序更是如此。如果不能确定多线程程序的执行行为的话，建议大家还是使用Java并发库、Pthreads等提供的锁、atomic类型、条件变量等更安全的方式来进行多线程同步。这些机制的内部实现都保证了相应的顺序性语义（atomic类型和锁还提供了相应的原子性语义），只要我们保证多线程程序没有数据竞跑、死锁等Bug，这些同步机制就能保证多线程同步操作的正确性。P

参考文献

- [1] Herb Sutter. volatile vs. volatile. <http://drdobbs.com/high-performance-computing/212701484>
- [2] Brian Goetz. Java 理论与实践: 正确使用 volatile 变量. <http://www.ibm.com/developerworks/cn/java/j-jtp06197.html>
- [3] Java并发编程实践3.1.4节
- [4] MSDN. volatile in Visual C++. [http://msdn.microsoft.com/en-us/library/12a04hfd\(VS.80\).aspx](http://msdn.microsoft.com/en-us/library/12a04hfd(VS.80).aspx)



陈冠诚

瑞典查尔姆斯理工大学（Chalmers）研究生，主要研究方向为并行计算、计算机体系结构及分布式系统。个人博客为并行实验室（<http://parallellabs.com>）。

责任编辑：董世晓（dongsx@csdn.net）

我的工具箱

文 / 潘加宇

看了不少期“我的工具箱”，从各位达人的文章里学到了许多有用的东西。相比之下，本人就比较低科技和落伍了：无iPad，无iPhone，无博客，无微博，基本不上开心网、人人网，不会思维导图……不过在这里还是分享一些相伴我多年的工具，供大家参考。

建模和开发工具

我的日常工作主要是研究UML相关的软件需求设计技术，为软件组织和软件开发人员提供服务。过去为团队讲解UML建模技能时，我一般使用Rational Rose作为演示建模的工具。从2002年底Rational被IBM收购以后，风格发生了很大变化，所以从2007年起，我演示建模的主要工具变成了Enterprise Architect，简称EA。EA的使用风格和以前的Rose接近，而且个头又不大，很方便开发人员自行安装学习。有时也有开发团队提出用StarUML等开源工具给他们讲解，所以电脑上也会安装一些这样的开源工具备用。

其他软件开发方面的软件，如编码环境Visual Studio、SQL Server数据库等，我使用的应该和大多数开发人员差不多。这些千篇一律的软件工具就不说了，下面只介绍我工作中比较有特点的工具。

核心：UMLChina业务系统

我最常面对却没有办法和大家分享的工具——是UMLChina自己的业务系统。UMLChina网站表面上极其简单，背后的工作却相当复杂。UMLChina定位为从深度上挖掘需求和设计技能，一旦从深度上定位，就会发现要做的事情非常多。2005年，我开始感觉原有的手段已经无法应付越来越多的学员和事务，于

是自己开发了一套系统——最近看了37signals在新书《重来》里面也说到要“挠自己的痒处”。

经过这些年不断的改进和升级，结合Word和Outlook的宏，该系统已经变成了一套相当全面的工具箱。这套工具箱确实为我省下了不少时间，更重要的是，它和UMLChina的业务结合在一起，不需要请一大堆人，就可以保持UMLChina的运作，为开发团队提供更深度的服务。有时走在大街上接到电话，“潘老师，我是某某某”，我当场就反应过来他是谁，哪个公司的，什么时候为他服务过。我并没有专门记忆这些信息，而是整天用着这个系统，服务过的伙伴们的信息不知不觉地融进我的血液中。

人人都需要为自己的生活打造一套特有的系统。

看书：FinePrint、Google桌面

我的工作需要大量的阅读，所以我的电脑硬盘上放了60多GB的电子书，大多是PDF格式。显然，这些书不可能都有空阅读，只是带在身边随时查阅。对于一些需要认真阅读的书，我还是习惯于“低科技”——把它打印出来，拿着纸张看，边看边在上面写写画画，这样更能引发深度的思考。

为了减轻需要携带的纸张重量，通常我都是先对PDF切白边，再用FinePrint缩印成一半大小打印出来。我从1998年开始就一直在用FinePrint，虽然后来Word和Arcobat的新版本也提供了缩印功能，我还是觉得FinePrint最好用。

如果是出差，没有即时可用的打印机，我就会把文件拷进SD卡，放进电纸书上看。2007

年我入手了一个E-Ink屏幕的翰林V8，用它看了不少书。可惜的是，V8不能直接读PDF，必须先要将PDF打印到翰林的虚拟打印机，变成翰林支持的格式。去年，方正番薯网的同事送了我一个YamBook电纸书，可以直接读PDF，还可以智能去白边，省心多了。电纸书的缺点是屏幕不够大，翻页需要等待，不过现在各个品牌也在出9.7寸屏的电纸书，屏幕已经和32K纸一样大，速度也在不断改善。但是，还是没办法在上面写写画画！我只能在看电纸书时，旁边再放一台笔记本，在纸上记笔记。可能有读者说，为何不买iPad？屏幕显示原理不同啊！E-Ink是折射光，看着不容易累，而iPad是透射光。

不管是阅读纸书时的写写画画还是阅读电纸书时的笔记，事后我会录进电脑中。为了省下录入的时间，我也曾想过要买一款类似zPen的数字记录笔，在做笔记的同时保存到PDF文件中，不过后来听说这种笔使用体验目前还不是很好，也就作罢了。

有时看到报纸杂志上有比较感兴趣的信息，我会用手机把它拍下来，然后导入电脑，变成资料。这时，一款比较适合近距离拍摄的手机就很重要，我用的是诺基亚N79，它有500万像素的摄像头，微距效果很好。苦恼的就是快门声关不掉，在书店中拍照会被店员注意到。

那么多的电子书和资料，需要有一个良好的搜索工具，所以我安装了Google桌面。遗憾的是，Google桌面总是不能够对那几十GB的电子书作索引。如果要在这么多PDF文件中搜索内容，还得用Arcobat Reader或Arcobat提供的PDF搜索功能，可以精确地搜索某个目录下所有PDF文件的内文。

视频音频：维棠、格式工厂

网络上有许多国内外专家演讲或教学的视频音频，可以在线观看和收听。不过，网络是靠不住的了，我更喜欢直接把它们下载下来，转成MP3，放进我随身的MP4里。在睡前听一些演讲或教学录音，听着听着就入睡了，这样也能潜移默化地学到一些东西。

如果是国内网站的视频，比较好的下载软件是维棠，支持大多数国内视频网站的下载。国外的网站如YouTube，也有类似KeepVid等专

门的下载工具。

现在存储设备都很便宜，写这篇文章时，2TB的西部数据移动硬盘卖680元，32GB金士顿SD卡才卖110元。所以，碰到好的资源，留一份在自己身边成本也并不高。

各种媒体格式之间的转换工具，我使用格式工厂，它支持目前各种主流的音频、视频、图片格式的转换。

通讯：增强无线网卡，手机管理工具

我经常出差，如何能保证随时有网络是一个问题。3G的资费目前还比较贵，我的解决方案是REALTEK增强型无线网卡，获取WLAN信号比笔记本电脑自带的无线网卡要强许多倍。现在大多数的建筑都部署了ChinaNet、CMCC、ChinaUnicom的无线热点，费用约每天1元，用这些服务比3G要便宜很多，而且没有超出流量限制的后顾之忧。实在是没有这些网络用了，我就临时用手机连上GPRS暂时对付一下。

我平时使用两部手机，多普达C720W和诺基亚N79。我使用智能手机工作室来管理C720W，使用诺基亚PC套件来管理N79。不久前创新工场出了个豌豆荚，不过只是针对Andriod手机的管理工具。UMLChina的业务系统也有连接手机发短信的功能，当需要给一群人发短信时，我会使用UMLChina系统来发。这也是我为什么一直没有换手机的原因，要是换了另一个平台的手机，短信的开发接口变了，还得重新开发，当初琢磨怎样连上这两个手机发短信就耗费了不少时间。

工具是锤子

2008年我在《程序员》杂志上写过一篇《专注》，说了一个锤子钉子的比喻。专注一个领域作为钉子，然后把其他领域的技术作为锤子，把钉子钉进这个社会。对于我来说，钉子就是需求和设计建模技能，上面我介绍的这些东西就是锤子。优秀的锤子能让钉子钉得更正，钉得更牢。P



潘加宇

UMLChina首席专家，潜心研究和实践UML/UP相关技术的应用。

责任编辑：高松（gaosong@csdn.net）

Maven十年，来认识下

文 / 许晓斌

历史

可能你会觉得惊讶，Maven听起来还有点新鲜的东西，真的有10年历史？10年在软件开发领域可不算短，要知道Java也只有16年的历史而已。根据Maven的创立者Jason Van Zyl的介绍，Maven最早的原型代码是在2001年提交的，2003年它成为了Apache的顶级项目，2005年10月Maven 2发布，渐渐地在开源社区流行起来，大概又过了5个年头，也就是2010年10月份，Maven 3正式发布，这时Maven几乎已经成为了Java构建的事实标准。

Maven在国外一直平稳地发展，而被中国程序员接受应该是始于2008年，随后得到了越来越多的拥护者，如图1所示：

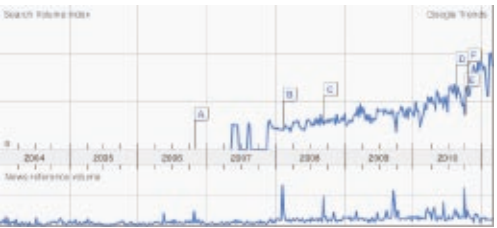


图1 从Google Trends可看出，Maven的接受程度逐年递增
我该用Maven吗？

Maven究竟是什么？该不该用它？回答了第一个问题后，读者再分析下自己的软件开发场景，就能得到第二个问题的答案。

对于Maven的一般理解是构建工具，就像用Make自动化构建Linux上的C语言程序，用Ant自动化构建Java程序一样。这个理解显然不完整，如果Maven是Java的构建工具？那我已经有Ant了吗？也有定义Maven为“项目管理工具”，这太模糊了，我给出的定义是BIR³D（注意R有平方），这是个非常好记的英文单词：

B表示构建，I表示信息聚合，两个R分别

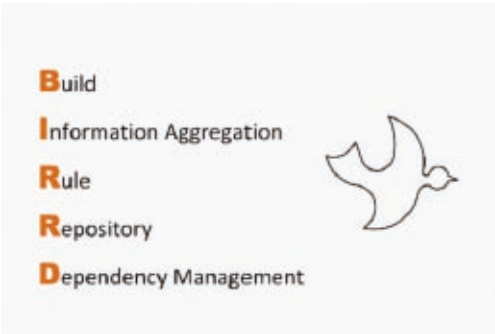


图2 用BIR³D定义Maven
表示规约和仓库，最后的D表示依赖管理。Maven所做的基本就是这些事情，下面来看看Maven是如何完成这些任务的。

自动化构建

编译、运行单元测试、打包等日常工作程序员再熟悉不过，重复的劳动应尽量被自动化，以节省程序员的时间。于是大家需要一个自动化构建工具，如Make、Ant、以及Maven，也会有人自己写Shell或者Bat脚本，其目的都是一样的。

Maven和Ant在构建方面有什么区别呢？主要在配置的方式上，Ant有build.xml配置文件，与之对应的，Maven有pom.xml配置文件。字面上可以略微看出点区别，使用Ant时，得告诉Ant具体想干什么。例如编译，得指定源码目录，编译输出目录，可能甚至还需要预先为这个任务创建好目录，这完全是过程式的。而使用Maven，只要配置基本的项目信息，比如项目坐标groupId、artifactId和version，Maven就知道如何编译了，这是声明式的。这当然没有魔法，Maven是借助文后将提到的规约来实现这种机制的。

使用过程式的配置方法，好处在于所有

的构建细节都一目了然，缺点是配置太多，且难以规范。使用声明式，好处在于配置十分简单，缺点是一旦出现了问题且你不理解内部机制，事情就会变得比较麻烦。

谈到构建不得不提Maven的生命周期模型，编译、测试、打包等等都对应一个Maven的生命周期阶段，这样的阶段是抽象的，Maven通过插件来实现具体的任务，例如maven-compiler-plugin被插入编译阶段来实现编译的任务。所以当你下载Maven的安装包的时候，你会发现它很小（3MB左右），相比之下Ant的安装包就大得多（超过10M）。

信息聚合

Maven得益于项目对象模型和自动化站点机制，能帮助团队方便及时地了解项目状态。例如要得到类似图3的测试覆盖率报告，只需简单配置一两个插件即可。



图3 通过简单配置插件，形成测试覆盖率报告

当然信息源还是必要的，例如为了生成变更记录报告，得告诉Maven项目的SCM信息，这都可以在pom.xml中进行配置。Maven所做的是帮助你以更好、更方便的方式呈现信息。

规约

没人喜欢受束缚，因此Maven所提倡的约定优于配置还是很讨一些人喜欢的，简单的来说，Maven希望能够按照它的约定来布局项目的源码目录、测试源码目录、Web资源目录等等。

Maven这么做一方面是为了实现“尽量少配置”理念，但最主要的目的还是降低沟通和学习成本。例如想参与一个开源项目，首先得签出源码构建之，如果它高度自定义了构建命令和目录结构，弄得构建项目很复杂，那显然会打击参与的热情。如果它使用Maven并遵守规约，那只需要执行 mvn clean install 命令就可

以。这也是开源项目如AppFuse、Struts 2使用Maven的原因之一。同样，在公司内部使用统一的规范也节省了不少人力成本。

仓库

如果开发环境没有机器能够连入因特网，那可能不应该使用Maven。Maven的核心几乎什么都没有，必须从仓库下载插件和依赖才能工作。这么做是有理由的，较之于从网站找下载链接，Maven用户现在只需要配置pom.xml就可以了：

```
<dependency>
  <groupId>junit</groupId>
  <artifactId>junit</artifactId>
  <version>4.7</version>
  <scope>test</scope>
</dependency>
```

Maven的中央仓库包含了绝大多数的开源类库，仅在2009年就接受了世界各地近400万单独IP访问，这个数字还在不停增长。如果担心无法在这么庞大的仓库中找到自己需要的依赖，还可以使用中央仓库提供的搜索引擎。

有时候会发现一些需要的依赖在中央仓库无法找到，这时最好的做法是建立本地的仓库服务器来代理包括中央仓库在内的所有外部仓库。中央仓库的维护者也在努力工作争取吸引更多的开源项目将他们的类库发布到中央仓库方便所有Maven用户使用。

依赖管理

前面提到，只需要配置XML即能获得并且使用所需要的依赖，这样非常方便，但比方便最重要的是清晰地明白自己所使用的依赖并对其进行有效地管理。有这样的场景：项目有名为lib/的目录，里面包含了少则几十多则上百的jar文件，各种类库，各种版本……暂且不说里面有多少完全没用的jar，一旦出现依赖冲突，在一堆乱麻中分析原因可不是件容易的事情。

Maven使得工作不再“靠巧合使用依赖”，用什么就配置什么，用户不仅知道用了JUnit，还知道这是JUnit 4.7。Maven还能防止同一依赖的不同版本被同时引入，它有默认的机制来自动选择，当然更好的做法是显式进行配置以避免不确定性。

Maven的“传递性依赖”概念也值得一提，例如使用Struts 2的时候，Struts 2自身也使

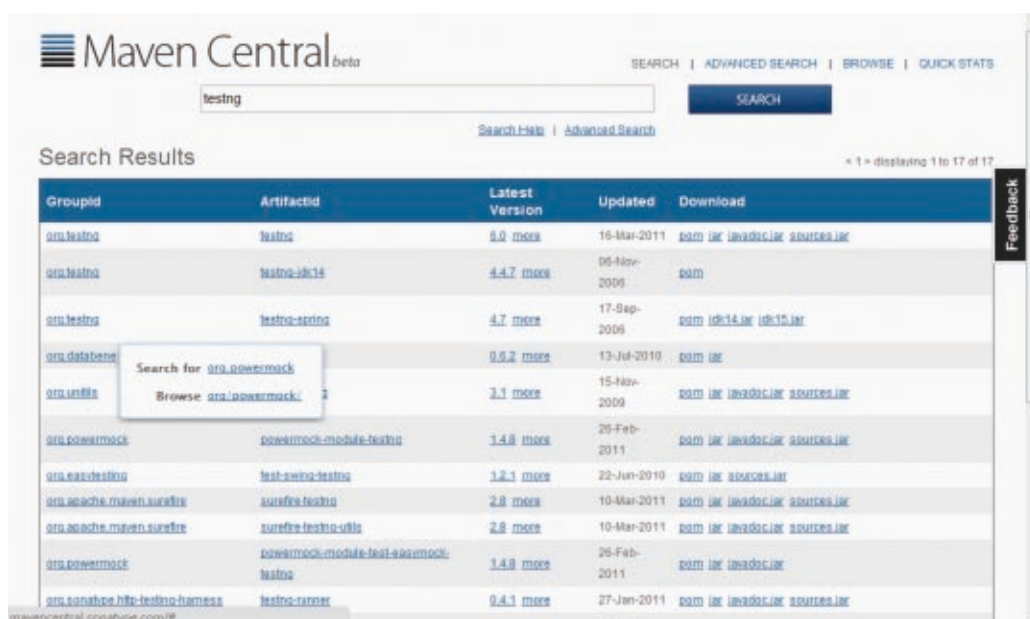


图4 利用Maven中央仓库提供的搜索引擎,可以找到所需要的开源类库

用到其他开源类库如OGNL，不过只要Struts 2的pom.xml配置正确，就无需关心OGNL，它会被自动地传递到当前项目的ClassPath中。

依赖管理是Maven非常核心的一个特性，这一点Ant的核心却没有，不过它发展了一个名为Ant Ivy 的子项目来提供类似的功能。

IDEA也有各自对应的插件。此外，Maven通过名为Surefire的插件，集成了JUnit和TestNG，几乎不需要配置，Maven就能自动运行单元测试。所有集成和扩展组成了围绕Maven的生态系统，让Maven用户的日常开发更加轻松。

围绕Maven的生态系统

如果之前了解过Maven，可能知道这些名词：Hudson、Nexus、m2eclipse、Artifactory、Surefire。其中有些和Maven并没有直接关系，例如Hudson，流行的持续集成服务器软件，使用Ant它照样能工作得很好，不过既然Maven是个出色的自动化构建工具，Hudson自然会针对它做集成，方便用户创建基于Maven的持续集成任务。其他持续集成服务器软件也有类似的特性。

Nexus和Artifactory是两款比较流行的Maven仓库服务器软件，有了这样的工具，可以在局域网内快速架设Maven仓库，代理外部仓库，也能存储内部上传的项目文件。Maven仓库服务器，或俗称Maven私服，能帮助用户节省外网带宽，大幅提高构建速度，再加上它能集中管理仓库，控制用户权限，毫不夸张地说，它是所有Maven用户的必备。

各个Java流行IDE也对Maven做了集成，例如Eclipse的m2eclipse插件，NetBeans和IntelliJ

综述

回到前面的问题：“该不该用Maven？”

本文是否已经解答了这个疑惑。一些知名软件公司在大规模使用Maven，但本质上还是要分析自身的开发环境，并仔细分析Maven的特性后，才能给自己清晰的答案。本文的目的就是介绍Maven自动化构建、依赖管理、信息聚合、规约、仓库等最核心的特性，当然Maven的特性并不仅限于此，例如它能通过Profile帮助你隔离构建环境，它能提供以多模块的方式管理项目，最新的Maven 3还引入了并行构建等很酷的特性。篇幅有限，这里不再详述。有兴趣的朋友可参考相关书籍。📖



许晓斌 (Juven Xu)

Maven技术专家，《Maven实战》作者，译著有《Maven权威指南》，就职于Sonatype，开源软件Nexus的核心开发者之一，目前主要从事Maven中央库相关的维护和开发工作。

责任编辑：高松 (gaosong@csdn.net)

非关系型数据库memlink浅析

文 / 冯勇 赵威

NoSQL系统种类繁多，从数据模型角度划分，有KV类型、文档类型、对象类型等；从存储方式角度划分，有行存储和列存储等。但有一种类型，业界很少提及，它就是Key→List类型，即Key指向的Value是List类型。这种类型很少提及，一方面是可被KV类型代替，比如，将List打包放入Value中；另一方面是大多数场景中采用SQL系统来存储List关系。

然而，还有一些特殊需求，上述方案不能满足，比如：海量论坛和海量微博服务。它们对List有下面两个独特的需求：

List是海量的。比如，百度贴吧的“李宇春吧”有超过2800万主题帖，李开复的新浪微博有超过400万粉丝数。这意味着为相应的数据模型，主题List有超过2800万元素、粉丝List有超过400万元素。

List中元素有动态排序需求。比如，贴吧的主题帖列表按最后回复时间排序，微博中的个人首页按微博发表时间排序。

如果再加上性能需求，使用传统的KV/SQL系统就很难解决了。稍有经验的开发者知道，使用KV系统的Value存储List，光打包解包就要花费很长时间，更不用说排序功能，SQL系统里上千万的数据order by也是非常缓慢的。

因此memlink应运而生！这是一款由天涯社区赞助的开源Key→List数据引擎，被应用于天涯来吧、天涯微博等产品。类似的产品还有Redis。而没有选择Redis，有以下几点原因：

- Redis数据持久化不够完善；
- 主从同步有缺陷，从服务器重启后，数据需要完全重新同步；
- 读操作为单线程，无法发挥多核CPU的处

理能力；

- 内存消耗过大。

介绍

如名称中的mem所示，memlink所有数据都建构在内存中，保证了系统的高性能（读性能大约是Redis几倍到十倍），同时采用块技术压缩内存，内存消耗比Redis小得多；持久化方面，使用redo-log技术；分布式方面，支持一主多从复制；此外还增加了数据过滤、读写分离、多核计算等工业级需求。

memlink支持三种List类型，其中，List中数据项顺序在插入时由参数指定；有序List的顺序按照数据项的值自动排序决定；Queue就是双向队列。

memlink客户端支持C、PHP、Python、Java等语言。核心的API详见<http://code.google.com/p/memlink/wiki/ClientAPI>。

设计原理

主框架

memlink采用异步I/O网络模型（基于libevent），由连接线程来管理大量并发连接，多线程处理实际命令。比起Redis单线程模型，memlink能很好利用服务器的多核特性。另外，因读写线程分离和多读一写模式，写线程拥有独立的端口和处理流程，即使在高并发的读请求下，写请求也能优先完成，符合大多数场景多读少写、写优先的特性。

核心数据结构

与Redis相似，memlink的核心数据结构就

是HashTable。但出于内存和性能的考虑，存储List上，memlink采用的是块链（Block），每N个结点构成块，块与块之间采用指针相连。这样处理的好处如下：

节省内存空间的使用。Block才有头指针和尾指针，不需要每个node都有。

遍历按块跳跃，提高查找效率。块链的大小是memlink的配置项，比如，在论坛中每页主题贴数量是50，则N可配置成50。memlink内部并不是所有的Block大小都是N，而是动态变化的。考虑这样的场景：我们有100万个Key，每个Key下的List只有1~2个数据项。这时如果分配的是N，内存浪费会非常严重。所以这里采用的是类似STL Vector动态增长的方法，动态增长到N，避免存在大量List的长度小于N时，造成空间浪费。

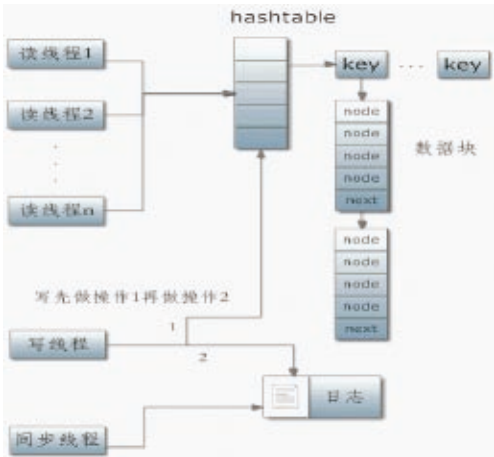


图1 memlink框架

memlink中的node大小由API指定，建议只存放id类型数据。每个节点还可附带一些简单的数字属性值。

为了提高并发性，memlink尽量不用锁。因为Block由内存池预分配，添加、修改node并不会造成内存非法访问，仅有可能某次读请求取到的是脏数据，但下次访问就是正确的数据。读操作远远大于写操作的情况下，读到脏数据的概率是很小的。未来如果有强一致性的需求，memlinke可以提供读写锁。

数据持久化

Redis持久化两种方式：第一、定时同步内存到文件系统中，这种方式不保证在两次同步

操作期间的数据的完整性；第二、将所有写操作命令记录到redo-log，重启时回放Log，这种方式由于Log大小不断增加，回放时间会越来越长。memlink同时借鉴了Redis这两种方式的优点，定期同步内存，在本次同步内存和下次之间采用redo-log保证数据的完整性。如果此期间memlink重启，会首先加载内存镜像，然后重放redo-log，这样就恢复到系统上次退出时的状态。

主从复制

memlink支持基于redo-log的一主多从的数据同步方式。从服务器会自动连接主服务器，并且发送同步的日志位置，主服务器将接着这个日志位置主动向从推送日志。这非常类似MySQL主从复制。

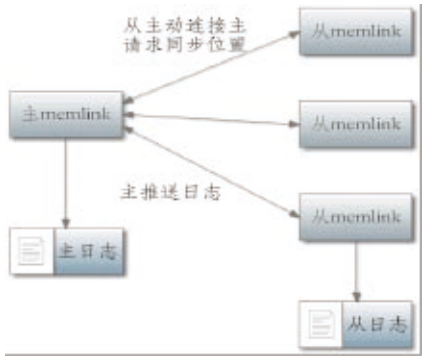


图2 memlink同步

性能对比

下面我们对memlink、Redis、MySQL进行简单性能测试对比，只针对insert和range。测试都是单机提供服务，且客户端在本机进行测试，并发50个短连接。测试一个Key下有1000万Value的情况（Key为5字节，Value为10字节），在0、1万、10万、100万、1000万的位置分别取100个Value。版本分别为Redis-2.2.2、MySQL-5.1.53，都使用默认配置。

硬件环境如下：

```
OS: CentOS release 5.4 (final)
Kernel: 2.6.18-164.el5 x86_64
Memory: 8G
Disk: 2T SATA
CPU: Intel Xeon E5405 2.0Gx2(4核x2)
```

数据模型如下：

其中，MySQL的数据表为：

```
CREATE TABLE TestList (
  listkey` char(5) not null,
  `listvalue` char(10) not null,
  `change_time` int unsigned not null,
```

```
key `testlist` (`listkey`, 'change_time')
)ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

MySQL查询语句为select listvalue from TestList where listkey=' \$key' order by change_time limit \$start,\$len

Redis命令为LRANGE \$key \$start \$end

memlink命令为RANGE \$key \$start \$len

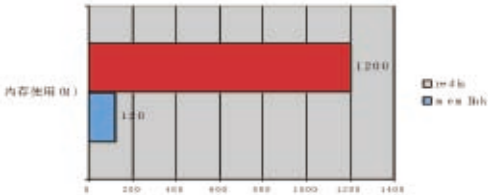


图3 内存消耗对比

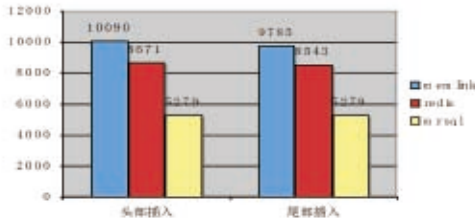


图4 insert性能对比(qps)

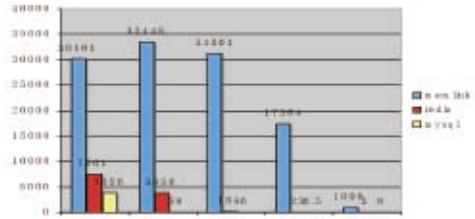


图5 range性能对比(qps)

使用场景

假设某款微博，日均发微博数2500万，活跃用户500万。下面是基于memlink的解决方案，供大家参考。核心的数据模型如下：

表1 对于微博产品中，核心的数据模型

我关注的人（关注列表）	key-list	上限2000
关注我的人（粉丝列表）	key-list	无限制
我的微博（feed列表）	key-list	无限制
我的首页（关注对象的feed列表）	key-list	有时间（15天内）和数量（500）双重限制
Feed id => Feed content	key-value	value为小数据块

微博主要的提交操作有添加关注、删除关注、发布微博。主要浏览操作有查看我的首页、我（他人）的微博、我（他人）的关注列表、我（他人）的粉丝列表。

本系统采用推模式将feed推送到所有粉丝，在大型微博产品中会采用推拉相结合的模

式。下面单机环境下的性能测试结果：（硬件环境和之前的测试一样），这里Key-Value系统使用的memcachedb。结果如表2所示（实现伪代码请参见《程序员》官网）。

表2 性能测试结果

操作	性能（qps）
添加关注（关注的人有100条以上微博）	1007
删除关注	3797
发布微博（有1000个粉丝）	149
查看首页	498
查看我的微博	502

memlink的特性限制

memlink是全内存存储，系统容量上限取决于系统的内存大小，因此正确的用法是只存储需要排序的ID数据。（Redis支持VM，即交换到文件中存储）。

因所有数据都存储在内存中，memlink对内存消耗比较大。List中的每个Value项是定长的，必须要占用固定大小的空间。每个Block也必须占用固定的node数。memlink中以value值作为条件来进行的操作（如delete、tag、mask、delete_by_mask），是逐个顺序查找的，因此这些操作性能上不会特别好。

未来的发展

memlink定位于高性能、持久化、分布式的Key→List数据引擎，并非存储模型的万能解决方案，没有计划去支持KV/Doc等其他类型。后面会加强对后面会加强对memlink管理、监控、统计、更好的分布式等方面的支持，可能增加新的List类型，以及非热点数据存储到磁盘等新特性。memlink将在天涯社区的一些产品中使用，成为天涯云计算的基础模块之一。



冯勇

前百度贴吧、地图产品架构师，前天涯高级架构师。



赵威

天涯社区系统平台部数据系统架构师。喜爱开源技术，关注高性能分布式系统，乐于技术交流和分享。坚持simple is better，简单就是美。

责任编辑：高松（gaosong@csdn.net）



Flashkus一次性闪存

对一般的闪存，我们希望外壳坚固、数据存储稳定，可以用很久，虽然最终仍不免沦为电子垃圾。来自俄罗斯工作室Art. Lebedev设计的Flashkus却反其道而行，他们认为闪存将演变为一次性的用品，因此设计成可拆分的硬纸板材质，材料易于回收。

GEEK 产品



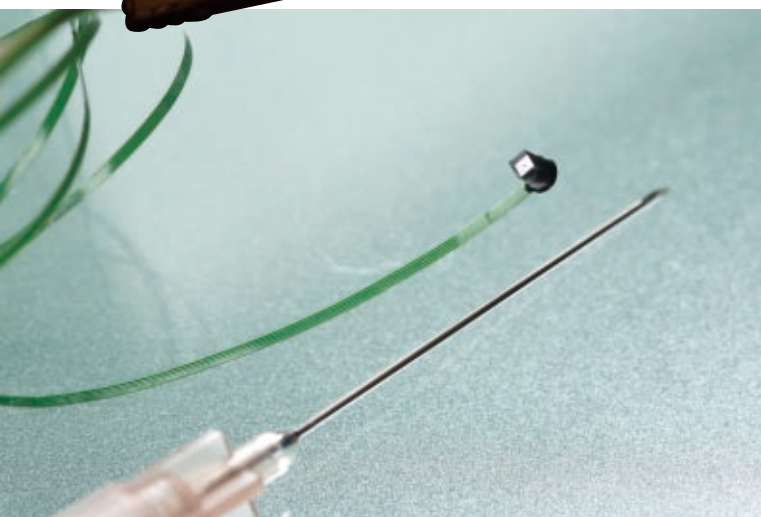
Abazhurus的自由组合灯罩

同样是由俄罗斯工作室Art. Lebedev设计的这款灯罩，基本素材是一组20块一模一样的塑料模板，模板上带有孔眼和卡槽，可以用来自由组合成不同形状的多面体。例如金字塔形的16面体、球形的20面体、椭圆形的16面体等。



愤怒的小鸟蛋糕

“愤怒的小鸟”在各大手机平台上的流行，带动了周边产品热卖。譬如这款以其游戏场景和主角为主题的蛋糕，应该会很受欢迎。国内的糕点房是否有这个创意呢？



Awaiba Camera

由Awaiba创造的这款摄像头可能是目前世界上最小的摄像头，整个大小只有1平方毫米，顶部有一个图像感光器，分辨率为250x250，可以通过线缆传输信号。

Cyber Snipa SONAR 5.1 Championship Headset

Cyber Snipa是欧洲专业做游戏外设的公司。最新推出的战术通讯设备Sonar 5.1，可以让玩家体验到真实的环绕音效。Sonar耳机拥有8个独立喇叭（每边4个，包含重音喇叭），音效软件支持喇叭独立式调整，可自定义虚拟喇叭位置以及提供多样化的预设音效。此外还包括噪声消减麦克风、可调式麦克风架设备，以及内建多媒体线控等特色。



Konect

也许是受了《星际迷航》等科幻电影的影响，国外设计师Olivier Demangel最近公布了一款名叫Konect的USB手表，外形科幻感十足，戴在手上颇有身处Tron或星际迷航世界里的感觉。除了时髦感十足的外形，这款手表还具备USB存储、音乐播放、蓝牙以及钟表功能。



新产品&新工具

开源操作系统 Cosmos

Cosmos全称C# Open Source Managed Operation System，是一款使用C#语言，基于.NET Framwork框架编写的开源操作系统。Cosmos的源代码使用C#编写，先把源代码编译成IL中间语言，再由专门针对该项目编写的编译器IL2CPU把编译好的IL源代码转换成本地机器码，目前处于里程碑5阶段，只支持x86体系CPU。



NAS服务器FreeNAS

FreeNAS是NAS服务专用操作系统。基于m0n0wall防火墙，该系统通过提供磁盘管理及RAID软件，可让用户将PC转换为NAS服务器。支持FTP/NFS/RSYNC/CIFS/AFP/UNISON/SSH协议，旨在让人们重新使用旧硬件。



新浪微博Java开发包 weibo4j

weibo4j又称sinat4j，是新浪微博的Java客户端开发包，支持Basic Auth、OAuth、上传图片、头像、地理位置等功能。

Google 发布针对 IE 9 的 WebM 插件

IE9正式发布后，已提供对视频的原生支持，但只支持H.264编码。但Google的WebM并非无缘IE9。Google推出了针对IE9的WebM插件，该插件支持Windows 7和Vista。

Oracle的Java移动开发框架

甲骨文发布了移动客户端和相关框架，以帮助开发人员快速构建工业移动设备的Java应用程序。该手机客户端只适用于Windows Mobile和RIM的BlackBerry设备。该框架目前还不支持iPhone、Android或Windows Phone 7。

转换图标小工具Ditaa

Ditaa是一款小巧的命令行工具，使用Java写成，能将ASCII字符画转化成图表。



应用Vitaa转化后，上面的文件变成：



SourceForge开源其平台软件

SourceForge在Apache许可证下发布了其平台软件Allura，允许开发者创建自己的开源平台。SourceForge的开发者将在博客上介绍Allura平台中Python、MongoDB、Solr和RabbitMQ的使用。

SAP BI 4.0和SAP EIM 4.0

历时三年时间，SAP Business Objects产品组合中的商业智能（BI）4.0和企业信息集成（EIM）4.0版本得以推出。它们具有以下特点：

- 借助内存计算工具HANA，能够快速处理大数据量，从而使实时商务成为可能；

- 通过任何移动终端设备，新的交互式可视化功能和用户体验增强功能使用户能随时随地完成工作，Sybase Unwired Platform为其提供了全面的移动BI套件；

- 深层文本分析整合了人们在社交媒体流、博客和电子邮件中所表达的想法和观点，并将其补充进传统的商业洞察中，这种新的多源、多维语义层及共同编程体验，简化了即时分析和内容创建过程，能以更快的速度向用户提供更全面的建议；

- 利用最新64位进制、虚拟化及云计算方法，提供直观的系统监视、预警及审计等功能。

之前获得授权的老用户，可以免费升级到4.0版本。

Java的JSON处理器 fastjson

fastjson是一款性能很好的用Java语言实现的JSON解析器和生成器，由阿里巴巴的工程师开发。其主要特点包括，快速（比其他任何基于Java的解析器和生成器更快）；支持普通JDK类（包括任意Java Bean Class、Collection、Map、Date或enum）；零依赖（除了JDK，没有依赖其他任何类库）。

起点R3企业级搜索引擎

起点R3企业级搜索引擎是起点软件提供的企业搜索解决方案，支持企业环境下的数据访问控制（ACL），能通过定义用户数据访问权限来控制检索数据的安全。



UltraMon 3.1.0

UltraMon是用于设置多显示器的工具软件，可以极大地提高工作效率，释放多显示器的全部潜能。

WordPress 发布云插件 Jetpack

WordPress可以说是目前最流行的开源Blog程序，其母公司Automattic为所有自主托管的WordPress用户提供了一个功能强大的云插件Jetpack。

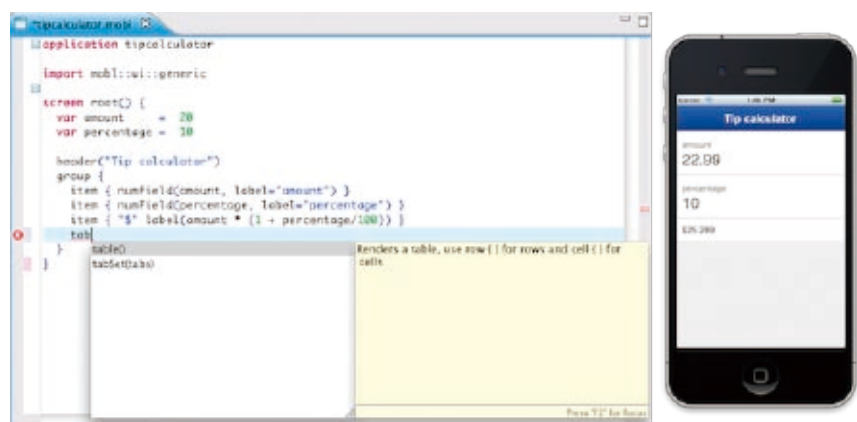
通过该插件，可以在自己架设的WordPress平台上使用WordPress.com上的相应功能，其中包括：

- 访问统计功能；
- 在边栏显示Twitter消息；
- 头像浮动卡片；
- 短网址；
- 分享按钮；
- LaTeX标记语言支持；
- 方便插入Youtube等网站上的分享代码等。

项目主页：<http://jetpack.me/>。

手机Web开发框架 mobil

mobil是一种新的开源编程语言，主要用于加速手机应用的开发，mobil可方便构建手机Web应用程序，可用于包括iOS、Android和其他支持HTML5技术的手机。





新书上架

ASP.NET本质论

作者：郝冠军

出版社：机械工业出版社

全书以ASP.NET应用中的请求处理过程为主线，深入剖析每一步处理涉及的技术和原理，同时列举和分析了开发过程中，各处理环节遇到的经典疑难问题，并给出解决方案。

书的前部分剖析了ASP.NET中的请求处理机制、对象与HTTP之间的映射关系、应用程序处理管道的处理过程、处理程序的处理机制，以及多线程技术在ASP.NET中的应用。在经典的WebForm部分，重点讲解了控件的原理与页面的生成机制。分析了ASP.NET MVC的处理过程，以及ASP.NET MVC应用与经典的ASP.NET应用间的关系。之后讨论了ASP.NET与IIS服务器之间的关系，并分别针对不同版本的IIS分析了处理过程。



HTML 5和样式应用指南

作者：Larry Aronson

译者：刘红伟

出版社：机械工业出版社

HTML被用来构建Web结构和编写Web内容，已经成为构建Web站点的标准语言。HTML5更是受到广大前端程序员的追捧，广泛应用已是大势所趋。本书包含了Web和HTML的基础知识、HTML语言的核心内容和各种重要元素的用法、对HTML实现样式化、HTML构建Web站点的技术和可能遇到的问题等。

作者结合五种流行的应用和服务（博客、GDocs、eBay销售、维基百科、HTML Email）一一介绍HTML标记的用法和特点，帮助读者实际应用HTML。本书还分别给出了各类HTML元素和CSS属性的详细介绍，其中包括最新的HTML5和CSS3的新增内容。书后附录还包括HTML5快速参考和CSS属性速查表。



iPhone与iPad开发实战

作者：Brandon Trebitowski

Christopher Allen

Shannon Appelcline

译者：张波 陈宝国 高伟 等

出版社：人民邮电出版社

本书由业界知名专家联合编写（Trebitowski为知名移动开发社区iCodeBlog.com的创始人；Allen是iphonewebdev.com的创办者；Appelcline则为在线游戏Skotos的运营总监），作者在更新已有iPhone应用程序的同时增加了最受追捧的iPad方面的内容。

除了介绍iPhone/iPad开发的基础知识（Objective-C和iPhone OS SDK；如何使用Xcode；如何使用Interface Builder；如何创建基本视图控制器；如何监控事件和动作；如何应用SQLite数据库等等）之外，还讨论了众多流行的主题，如加速计、点对点游戏、Map Kit框架、推送通知和应用内购买功能。全书通过逐步指导，教你如何将API同新应用程序或者已有的应用程序整合。



大国、民企、正道

作者：洪峥

出版社：企业管理出版社

此书以全新的视角解读我国民营企业发展的关键要素问题，提出了企业创业发展全过程的理念，以及企业发展要与投融资和资本市场相结合，是为民营企业发展壮大的正道。书中全面涵盖了企业发展的全过程，深入浅出地阐述了企业“短板”的解决办法，既有理论深度，又有实战操作性。

如果你计划创业，此书会指导你创业全过程、发展路线图、资源配置策略。

如果你正在创业且不顺利，此书会帮你理清发展思路、如何有效整合配置资源。

如果你的企业正在走向做大做强阶段，此书会使得你重新认识创业难，守业更难，打造“百年老店”企业更具使命感的正道。



Web表单设计: 创建高可用性的网页表单

作者: Caroline Jarrett Gerry Gaffney

译者: 董淑桢

出版社: 人民邮电出版社

网页表单无处不在——从注册表单到联系信息表,从商业领域到政府部门。成功的表单设计不仅能提高用户满意度,还能收集更加精确的数据并降低维护费用;而失败的表单设计不仅会收集到与需求相悖的冗余信息,还极有可能导致潜在消费者的流失。

设计具有高可用性的表单绝非易事。作者明确阐述了如何从表单的关系、对话和外观三层模型出发,设计出具有高可用性的优质网页表单,并通过可用性测试及早发现表单的潜在问题。以及如何定义需求,如何提出与用户期望相符且容易理解的问题,如何撰写说明文字、设置进度指示器和处理出错信息。

全书包含从细节展现(如标签对齐方式、必填字段)到视觉设计(如创建整齐的网格、颜色应用等)等大量实例。



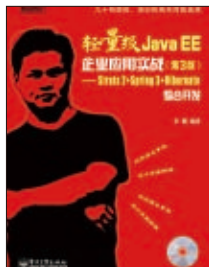
轻量级Java EE企业应用实战——Struts 2+Spring 3+Hibernate整合开发

作者: 李刚

出版社: 电子工业出版社

本书介绍的开发平台,是以Struts 2.2+Spring 3.0+Hibernate 3.6为核心的轻量级Java EE,这种组合在保留经典Java EE应用架构、高度可扩展性、高度可维护性的基础上,降低了Java EE应用的开发、部署成本,对于大部分中小型企业应用是首选。在一些需要具有高度伸缩性、高度稳定性的企业应用里,以EJB 3+JPA为核心的经典Java EE应用则具有广泛的占有率。

作者采用最新的Tomcat 7作为Web服务器,详细介绍了Spring和Hibernate的“零配置”特性,并充分介绍了Struts 2的Convention支持。书中不仅介绍了Spring 2.x的AOP支持,还详细介绍了Spring 2.x中Schema配置所支持的util、aop、tx等命名空间,最后简要讲解了AspectJ的相关内容。



网站重构: 应用Web标准进行设计

作者: Jeffrey Zeldman Ethan Marcotte

译者: 傅捷 祝军 李宏

出版社: 电子工业出版社

畅销书作家、设计师、网页标准教父Jeffrey Zeldman再次更新了他的经典著作。这已经是本书的第3版,此次更新基本涵盖了随着环境和技术的变化,Web标准所面临的挑战以及因此而发生的改善。

一如既往,本书提供了更多明确的见解和新的例子,来阐述基于标准的设计的核心思想。这不是一本提高技术的书籍,而是一本改变观念的指南。

除了有很多的“为什么”来提出问题,本书还提供了大量的方法来解决这些问题。在写作上,作者轻松的笔调,使复杂的技术更易于理解消化,帮助读者轻松创建一个访问快速、低成本维护和开发、可承受更多用户的访问网站,并把页面内容和表现方法相分离的理念,更深刻地传达给读者。



全球排行榜

Amazon	
1	Search Engine Optimization (SEO) Secrets
2	Don't Make Me Think: A Common Sense Approach to Web Usability, 2nd Edition
3	Cracking the Coding Interview, Fourth Edition: 150 Programming Interview Questions and Solutions
4	Head First Java, 2nd Edition
5	Programming in Objective-C 2.0 (2nd Edition)
6	Beginning iPhone 4 Development: Exploring the iOS SDK
7	Head First HTML with CSS & XHTML
8	Working at the Ubuntu Command-Line Prompt (Linux Nitty Gritty)
9	Presentation Zen: Simple Ideas on Presentation Design and Delivery
10	C Programming Language (2nd Edition)

天珑书局 (中国台湾)

1	Android案例开发完全讲义
2	鸟哥的Linux私房菜-基础学习篇, 3/e
3	深入浅出Android游戏程序开发范例大全
4	Google! Android 2手机应用程序设计入门, 3/e
5	深入浅出Android系统原理及开发要点
6	最严选PHP案例模块开发讲座
7	iPhone创意程序设计家, 2/e
8	Google Android SDK开发范例大全2
9	信息架构学网站应用, 3/e
10	App程序设计入门: iPhone & iPad

第二书店

1	ASP.NET本质论
2	设计原本: 计算机科学巨匠Frederick P. Brooks的思考
3	深入理解计算机系统
4	项目百态: 深入理解软件项目行为模式
5	Android系统级深入开发-移植与调试
6	构建高性能可扩展ASP.NET网站
7	算法导论
8	Android应用开发揭秘
9	Erlang编程指南
10	C程序设计语言



刘龙龙，台湾铭传大学国际学院教授。

从软件复用看现代程序员的竞争力

软件复用是软件开发的技术问题，着重如何能有效提高程序员的生产力与软件的质量。虽然项目管理人员都非常期望程序员多多善用软件复用的技术与概念，但仍有不少程序员喜欢自己编程而不采用别人已经完成的软件。现代职场上，企业习惯考虑的因素是“竞争力”而不是程序员自己编程后的成就感。程序员彼此之间互相切磋，也会互相较劲，维持大家共同进步是一项传统；可半世纪以来，前辈程序员们累积开发的软件种类及数量，如今已是非常充裕，需要的任何软件几乎都可取得，而且质量也都非常优良。那么，年轻辈的程序员要怎样维持大家共同进步、又具有一定水平的“竞争力”呢？

首先看一下软件复用的含义，其概念主要源自计算器与电子产业的成功发展模式。从最早期的电阻、电容，到印刷线路版上的集成模块、现代的芯片等，计算机硬件都是由有规格的零组件设计并组装而成的。那么，软件也如法炮制应该可以得到同样的效果。可惜，软件毕竟与硬件完全不同，尤其两款软件间的接口描述与制作无法像硬件那样明确，因此成效并不理想。一般而言，优良的软件是不容易被开发出来的，所以一旦有好软件在程序员间流传开来，大家就会保留一份以便参考或是直接使用。渐渐地，例如编译程序、输入输出函数库、数学计算函数库等就成为程序员必备的软件开发工具。

一段优良的程序码，多是由经年累月的测试、使用、更新逐渐形成的，有着可被追踪的历史背景与颇具火候的耐用度，才会有一定的质量水平。可以说，它是某个天时、地利、人和的产物，要把相同的条件搭配起来，机会很低微。经常会发生没有经验的程序员把自己的代码误改，想要改回去却没有办法。除了程序之外，软件复用技术可以延伸到系统架构、需求分析，乃至软件开发方法。

三十年前的程序员与现代的程序员在工作型态上最大的不同是以前什么都缺、程序员要自己搞定所有的软件；而现代则是什么都有，但程序员要去找自己需要的软件。以前程序员所切磋、较劲的，是计算方法与数据结构，而现代程

序员面对的则是数据挖掘、云计算。前者需要数理基础、思考逻辑明确，后者需要创意整合、符合时尚需要。两者面对的软件规模方面差异很大，以前两三千行的程序就是大型编程，现在几万行的软件制作还算平常。在生产力指数方面，人月所产出的有效程序变化不大，只是以前是以汇编语言为计算单位，后来以高级程序语言为计算单位，现在则以服务描述语言为计算单位。在质量方面，似乎早期的编程技术比现代要细致得多，有许多知名软件的核心程序部分仍然依据非常早期的基本编程技术性，而扩充的部分则多偏重在处理美观以及增加可使用度等功能。

有人说，现代软件开发项目中软件复用的比例已经可以达到98%，只有极为优秀的程序员才有机会参与2%的新软件开发工作（据说这些新软件都是属于未来人机界面的应用）。如果这是真的，现代程序员可能就必须再多想想怎样积极采纳软件复用。Google App Engine在最近的推广文宣中就明白表示服务提供商不需要会Java或Python编程技术就可以利用它的云计算把软件搞定，信不信由你。它已经把软件复用技术发挥到这个地步了，所以程序员自己若要执意编程、开发软件，只是当成一种嗜好或活动没有问题；但是若要当成日后主要谋生工具，可能太过于乐观。这个论述绝对不是说程序员的未来出路不好还是怎样，而是提醒大家现代程序员如果想法及做法还停在二十年前的水平，那一定没有竞争力的。换个角度说，现在的编程技术与测试技术在职场需求上相比，哪一个比较重要？还有，软件工程相关技术中，编程技术占多少比例？其他还有哪些待程序员投入的技术？

软件复用与知识产权，两者都是值得尊敬的。程序员在了解软件复用的意义及现状之后，还要清楚了解知识产权的意义才不会闹笑话。简单以程序为例来看，软件复用不是代表鼓励盗版或抄袭，而是要合法得到用户许可证。如果这个授权是开源码GPL授权，软件复用就遵循授权内容使用；如果这个授权是一般付费条件授权，那就付费授权。请再次记得，优良的软件是不容易被开发出来的。🔒



高巍，安卓爱普公司创始人。原搜狐媒体产品中心经理。关注移动互联网、互联网产品管理。
微博：<http://t.sina.com.cn/inetpm>

长翅膀的不一定是天使

“长翅膀的不一定是天使，也有可能是鸟人；骑白马的不一定是王子，也有可能是唐僧。”这是一句网络流行的调侃语录，放在当前炙手可热的互联网创业与投资的语境下，戏谑中也有几分真实。

中国不是美国，中关村也不是硅谷。中国每年新创企业据说是美国的十倍，但天使投资却占不到新创企业总数的1%。创业人群与天使投资之间的供求关系完全不对等，天使投资人太少，想找钱的创业者太多。在中国，天使投资看起来强势，创业者看起来相对弱势，时有创业者与投资相互被欺骗的事件发生，如天使投资人何伯权所言，根源就是天使投资人与创业者之间的供求关系极度不平衡。目前阶段，创业者与天使投资人都要交学费，学费交了就逐渐成熟起来了。

首先，找什么样的天使投资？知名投资人挑选项目的余地和话语权都很大。手机游戏领域的创业者吴刚认为，“就成本和性价比来看，我觉得很多不出名的、业绩好的（投资人），倒很有点儿意思。创业者可以多去找找这些人，一个很好的投资人告诉我一句话，推动资本前进的，不是浪花，而是暗涌。”而兰亭集势的创业者郭去疾建议，“投资人风格与你是否搭调永远排第一位。”投资进来的钱都是有个性，你必须理解钱后面的人。双方的沟通、信任、包容，甚至理念、人生观，大的战略方向是否一致？如果不对路，那就是漫长痛苦的开始。经纬创投的张颖建议，“谨慎拿钱，哪怕选择有限。”可现实情况是，创业者迫于种种压力，往往只能选择到账最快的投资。

接下来，就是敏感的估值和股份比例。“股权压榨”是创业者圈子中抱怨较多的。雷军在一次演讲中笑把风险投资比喻成高利贷，称天使投资可能是资本链条里“最黑”的一段。他对此的解释是天使投资人承担着巨大的风险，而举出此类比喻的目的，是希望创业者在融资时能够认真了解不同投资人对于回报的预期。早期融资的天使阶段，项目生死两茫茫。VC是把相对比较清晰的业务快速放大，风险相对小；而天使投资人要和创业者一起探索未知，要承担比较高的项目灭顶风险。

被投资的企业倒闭时，通常最优先清偿的是最后一轮的投资人（例如第二轮），再来是上一轮投资人（例如第一轮），然后是天使投资人，最后才是创业者。天使投资的风险是仅次于创业者的。另外，创业者本人对预期成功的自信满满，对项目的敝帚自珍的估值，在早期其实相当不靠谱，实际运作跟计划书很可能是两回事。一位创业者的建议是，从价值比例的角度来考虑天使投资的股份比例是误区，创业者并非出卖项目的一部分，而是引入一位“有钱出钱”的创业伙伴。卖股份还是引伙伴，这是两个完全不同的角度，会导致完全不同的心态和行为。总之，别后悔、别反悔、别不平衡。

然后是投资条款。比较珍惜名誉的天使投资人不会在合同上做小手脚，传出去实在不好看。但也不排除一些“堕落天使”，在投资协议中设计一些“我抓住你的把柄了”的条款，或者准备一些“杠杆”来取得控制权。郭去疾有三点建议：估值只是条款，真正玄机在其他地方；雇一名好律师；拒绝对赌。对某些过于苛刻的条款，易凯资本王冉评价为“没事找抽”型，“你要什么风险都不承担还想赚钱，那不明摆着是等着让被投资人背后算计你？”

最大争议还在投资后，对企业管理和运作的介入程度。某些天使投资人，如徐小平，主张“好的天使投资人投完就该忘记，然后做创业者最好的伙伴，就像太阳与空气，默默陪伴。”有的则会相对强势介入，甚至亲自披挂上阵，我知道的最近人事动荡比较大的一家创业公司，中层经理的周报邮件都要抄送投资人。创业者本人的个性与“断奶”程度、成熟程度，也决定了对天使投资人的依赖程度不同。有的创业者说，最怕天使投资人有“事业心”，创业者某种意义上为了拥有股份而已的职业经理人。也有创业者，如世纪佳缘龚海燕，谈到天使投资人钱永强时，“有他跟我在一起的话，我不用那么担心，我觉得他可以看得更清楚一些。”

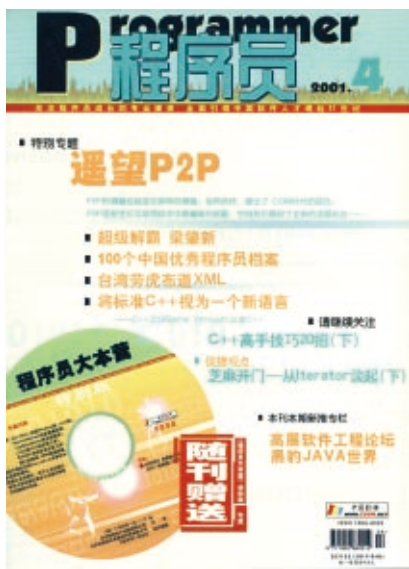
创业者与天使投资人，是一小部分人与另外一小部分人的神奇相遇，2%有梦想的人遇到了另外2%支持他梦想的人。创业者有感恩之心，天使投资人有包容之心，才成就了一个个改变世界的梦想。P

10年前： 《程序员》2001年4期

IT业无疑是近10年变化最快的行业了。10年前互联网方兴未艾时，恐怕鲜有人想象得到如今的移动物联时代。如此眼花缭乱的变化里，有无可供借鉴的规律？《程序员》从本期开始，开辟“10年前”专栏，盘点10年前的同期期刊重点报道的人物、技术、商业后来的兴衰变化。希望这样的尝试，能给关注中国软件的读者们带来一些新的视角。

从2001年4期《程序员》杂志的封面就可以看出，当时的中国软件业正处于如火如荼的共享软件时代，这是诞生了程序员个人英雄的黄金岁月。翻开杂志的重点报道《100个中国优秀程序员档案》：求伯君（WPS）、王永明（五笔字型）、陈虎（智能陈桥）、洪以容（网络蚂蚁）、张小虎（Foxmail）……他们中有的至今星光璀璨，有的则销声匿迹了。历史证明，决定他们人生起伏的关键，不是技术，而是他们对于技术与商业联系的理解。

本期的封面人物是豪杰解霸的作者梁肇新。作为老一辈程序员英雄的代表，梁肇新的奋斗历程里的抉择与彷徨、成功与失败是那一代技术精英们的缩影。1993年，刚从高校毕业的梁肇新去珠海的巨人集团求职，一个细节打动了招聘官：他随身带的厚厚六本笔记本，抄满了关于MINUX的10000多行代码。当时没有互联网，计算机图书也不兴旺，老一代的程序员们就是靠如此看似笨拙的苦功夫，实现自己的技术蜕变。随着他在金山影霸、豪杰解霸上功成名就，开始创业构建一个团队时，一个编程奇才的软肋立刻彰显了。记得他说：“我招聘了一些开发人员，一开始我不知道该让他们干什么，就让他们自己研究。但这样一来，感觉很长时间都没有真正做出什么东西。后来我又尝试着让开发人员每天写文档，还是不行。到现在才明白那都是没有一个良好的管



理机制。”结局是，梁肇新经常又当将军又当士兵，一次又一次亲自解决手下碰到的技术难题，自然团队效率可想而知了。于是梁肇新步履蹒跚地开始团队管理的探索，比如钻研起了CMM。

总体说来，梁肇新一辈的程序员，对商业的理解，属于摸着石头过河探索的阶段。当时的杂志执行主编李学凌先生在题为《超善恶》的卷首语中，对于“共享与商业”的关系进行了现在看来颇具前瞻性的思考。他说：“如果没一个编写共享软件的人都不严格按照商业的计划去运营和发展自己的软件，就会遭遇失败。我们现在过多地关注了成功的共享软件，还是有更多人浪费自己的才华的时间，结果半途而废。商业在更大尺度上，代表着责任。”

当时重点报道的热点技术是P2P，它的模式、理念与共享的精神一脉相承。在P2P的架构下，不再是传统的服务器\客户机层级架构，所有网络内的工作站都具有级别相同的工作能力。在4期杂志《P2P：I2时代的革命先锋》一文中，激情畅想了P2P能带来的新局面：超级计算、协力工作、超级搜索、文件交换（分布式存储）……它和今天的“云计算”多么相似！而当时的担忧，普遍集中在P2P带来的“互联网的无政府主义”、版权保护、信息安全等方面的隐患。10年过去了，2009年底，中国第一P2P门户BTChina分崩离析，而不久前著名P2P分享网站VeryCD，突然停止了影音下载等服务，其创始人黄一孟在微博上说：“7年的心血和积累，说关就要关，说停就要停，没有人能甘心……”P2P，以及它所代表的“互联网内容共享黄金时代”，在中国似乎已经夕阳尽头，着实让人唏嘘不已。

2001年的梁肇新说“在我的理想中，豪杰公司像中国的微软……”事实上，那一辈的程序员英雄们没有一个能成就为“中国的微软”，而这10年来，他们心目中的榜样——“世界的微软”也同样屡失发展良机。看来成功的IT逐浪者，未必是技术最好、资本最雄厚的，却一定是最擅于把握时势的。据悉，梁肇新目前正在做面向家庭和小放映厅的3D电影解决方案，他认为中国电影票房过10亿的时代即将到来。P



Robbin: 
Unix程序员, AppleFans, Geek, 急需女朋友。命令行狂热者。

Hank: 
Windows程序员.NET高手。已婚。保守派。

Ada: 
架构师。第三性别。喜欢挖苦男人。热爱无所不能的emacs。

作者介绍: 西乔
项目经理。06年起携创业团队从事Web技术外包开发及产品咨询顾问。



如果你有什么好玩的关于程序员的故事、对话、代码，愿意通过漫画的形式分享，请给西乔发邮件：arthur369@gmail.com。

幽默

为了程序没日没夜研究学习，为了程序整天灌输OO思想，为了爱好拼命敲着代码，张口闭口类和对象，这些都值得吗？到头来把对象给实例化了，自己的对象没找着……

原来我以为“我的女友”是一个普通类，可以构造出多个实例……直到有一天，我才发觉她是由单例模式设计，实例返回永远只能是你……

老婆的应用非常广泛，但这并不是简单的概念，背后的数据结构非常复杂。把她当作基本的数据类型来对待，是非常错误的，这样做往往会引起不可知的后果。老婆经常被称为“对象”，但在某些系统（比如某些摩门教系统）中，老婆可能是对象数组。

孩子哭闹的时候就在边上打开吸尘器，然后拍小孩的背，一会儿小孩子就睡着了。理论依据是小孩子最多同时关注两件事情，如果有三件事情需要同时关注，他就只能睡觉了。结论：所有小孩都是单核处理器，一旦多线程，就自动挂起。

P.S.: 这爹是个程序员！

你编，或者不编程，项目就在那里，还未完成。你调，或者不调，Bug就在那里，早晚得改。你烦，或者不烦，老板就在那里，唠叨不休。你改，或者不改，需求总在变化，不改不行。你来或者不来，打卡机就在那里，爱来不来。默然，编程。寂寞，难耐……

一位失眠的程序员是这样试图睡觉的：1只羊、2只羊……32766只羊、32767只羊，-32768只羊，-32767……天啊！这个变量为什么是short！

据说一老外年轻时，立志要当一名伟大的作家。怎么才算伟大呢？他说：我写的东西全世界都要看到！看完他们必定会歇斯底里！会火冒三丈！会痛苦万分！结果，他成功了，他在微软负责写系统蓝屏时的报错提示信息……

有一天，Windows进度条编程工程师去拜访他的朋友。边走边打电话告诉朋友什么时候到：我已经出门了，正在往你那里赶，大约15分钟后到达……大约1小时后到达……大约7天后到达……大约7秒后到达……